PC-9800シリーズ

FM音源スーパーサウンド



窪田 弘　田中寿郎　広瀬 真

SHUWA SYSTEM TRADING CO.,LTD.

PC-9800シリーズ
FM音源スーパーサウンド

窪田　弘

田中寿郎

広瀬　真

Shuwa System Trading Co.,Ltd.

ヤマハのシンセサイザ、DX／TXシリーズに搭載され、さらにパーソナルコンピュータPC－8801mkIISR、PC－8001mkIISR、PC－6601SR、PC－6001mkIISRに搭載されてコンピュータミュージックの可能性を大きく広げたFM音源が、ベストセラーマシンPC－9801シリーズにも搭載されました。PC－9801UV2では標準実装、その他の機種でもサウンドボードを実装すれば、FM音源を自在に楽しむことができます。オリジナルな音色や曲作りにはもちろん、最近はゲームなどにもFM音源に対応したものがでてきていますから、迫力満点です。

FM音源は従来のPSGとは異なり、各種の楽器音はもとより効果音、さらには人声まで、ほとんど無限ともいってよいほど多くの音色を表現できます。しかも、アナログシンセサイザのように高価なオプションモジュールを購入したり、パッチコードを張り巡らせたりする必要はなく、非常にコンパクトにまとまっています。

しかし、FM音源を扱うには、それなりのテクニック、ノウハウが必要です。なぜFM音源がこのような高い機能を持つかといえば、それは従来とはまったく違う原理によって音を創り出しているからなのです。FM音源の音作りには、アナログシンセサイザのノウハウはもはや役に立ちません。本書は、FM音源による音作りのノウハウを、できるだけ提供することを目的としています。

本書は、以下のような構成をとっています。

第1章　FM音源LSI　YM－2203について、そのハードウェアと操作法について説明します。やや難しい内容ですので読み飛ばしても構いません。

第2章　FM音源による音作りの基本として、FM音源の各パラメータについて、それが音色に対してどのような効果を持つかということを説明します。

第3章　PCシリーズに内蔵されているすべてのプリセットボイスについて解説します。

第4章　音作りの応用として、プリセットボイスのエディット、オリジナルボイスの作成、さらにヤマハのDXシリーズからのデータ移植法などを解説します。

付録　PCシリーズのサウンド機能に関する資料を集めてみました。

そもそも、音というものについて文章で語ることには無理があるかもしれません。これは、本書ではもっとも苦労した部分です。パラメータをどう設定するとどういう音がでる、と書くのは簡単ですが、読者に音のイメージが伝わらなければ、本書の目的は半分も達成されないからです。

本書が読者貴兄のサウンドライフの充実に役立つことを願ってやみません。末筆になりますが、本書の企画・発刊にあたって多くの方々にご助力をいただいたことを、この場を借りてお礼申し上げます。

1986年11月　　著者しるす

第1章 FM音源の原理 11

第4章　音創りの応用　135

付　録　197

本書を読むにあたって

○　本書は、ＰＣ－９８００シリーズ本体とサウンドボードの組み合わせをＰＣシリーズと略称しています。ただしサウンドボード標準実装のＰＣ－９８０１ＵＶ２は、本体のみの名称です。

○　本書の内容と対応する機種について解説します。

第１章　FM音源ＬＳＩ　YM－２２０３について解説した章です。

第２章　FM音源による音創りの基本について述べた章です。

第３章　内蔵されているすべての音色データについて解説した章です。

第４章　音創りの応用・実践編です。

第５章　　知っておくと便利な知識・テクニックを紹介します。

○　プログラムはMS－DOS版N88－日本語ＢＡＳＩＣ（86）ver3.0で作成しております。よって、MS－DOS版でないN88－日本語ＢＡＳＩＣ（86）では、ＣＬＥＡＲ文を変更する必要があります。

具体的には次のようにして下さい。

```
××××　　ＣＬＥＡＲ　　&Ｈ×××
              ↓
××××　　ＣＬＥＡＲ　　,&Ｈ４０００
```

なお、MS－DOS版ＢＡＳＩＣでサウンド機能を使用する場合、

```
n88basic　　/e:sound
```

としてＢＡＳＩＣを起動させる必要があります。

なお、第１章および第５章は田中が、第２章および第４章は窪田が、第３章は広瀬がそれぞれ担当しました。

本書を読むにあたって

○ 本書は、PC－9801シリーズ本体とサウンドボードの構成で動作する「FM音源」について解説しています。ただしサウンドボード標準装備のPC－9801UV21は、本体のみが対象です。

○ 本書の内容と対応する機種について解説します。

第1章 FM音源LSI YM－2203について解説しています。

第2章 FM音源による音創りの基本について述べています。

第3章 内蔵されているすべての音色データについて解説しています。

第4章 音創りの応用・実践編です。

第5章 知っておくと便利な知識・テクニックを紹介します。

○ プログラムはMS－DOS版N88－日本語BASIC（86）ver.2.0で作成し、動作確認しました。よって、MS－DOS版でないN88－日本語BASIC（86）では、CLEAR文を変更する必要があります。

具体的には次のように変更します。

```
×××  CLEAR  ,&H×××
        ↓
×××  CLEAR  ,&H4000
```

なお、MS－DOS版BASICでサウンド機能を使用する場合、

```
>N88basic /e:sound
```

としてBASICを起動させる必要があります。

なお、第1章および第5章は田中が、第2章および第4章は住田が、第3章は2人で分担して執筆しました。

第1章

FM音源の原理

FM音源の原理

1.1　FM音源の原理

シンセサイザで音を創るとき、考えなければならないものに、”音の３要素”があります。それは、”音の高さ（周波数）”、”音色（波形）”、”音量（振幅）”の３つで、すべての音はこの３つの時間的変化によって表わすことができます。

このうち、音の高さと音量については、電気的な方法によって制御することができます。しかし、音色についてはそれほど簡単ではなく、さまざまな工夫が凝らされています。シンセサイザにはいろいろな方式がありますが、それは主に音色の制御方法の違いによるものです。

FM方式によるシンセサイザは、Frequency Modulation（周波数変調）によって音色を制御します。そのようすを少し具体的に解説しましょう。

図１－１において、ＶＣＯはサイン波を発生する発振器で、入力電圧によって出力するサイン波の周波数が変わるものとします。図１－１(a)のように、プラスの電圧をかけると高い周波数のサイン波を、(b)のようにマイナスの電圧をかけると低い周波数のサイン波を出力します。

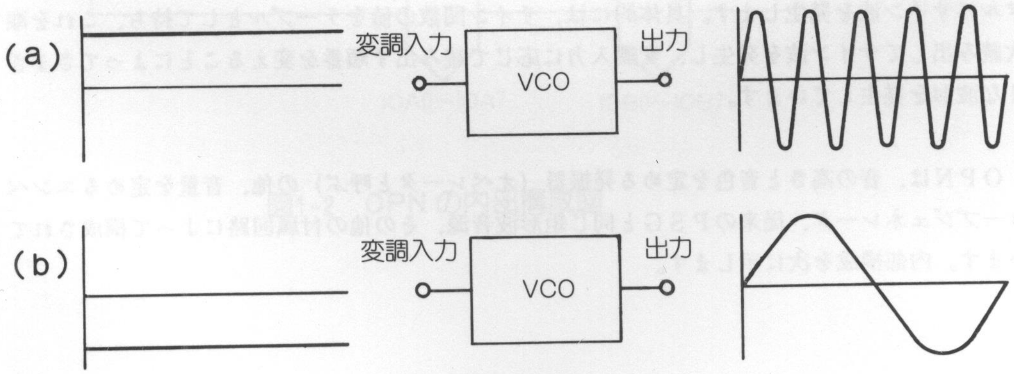


図1-1　周波数変調　その1

さて、このようなＶＣＯに、(c)のような方形波を入力します。すると、(a)と(b)の周波数が交互に出力されて、図に示すような出力が得られます。

つまり、サイン波が方形波によって周波数変調を受けているということになります。　さらに周波数の高い方形波によって変調をかけると、(d)のようになります。こうなるともはや出力はサイン波とはいえず、波形が変化したということになります。

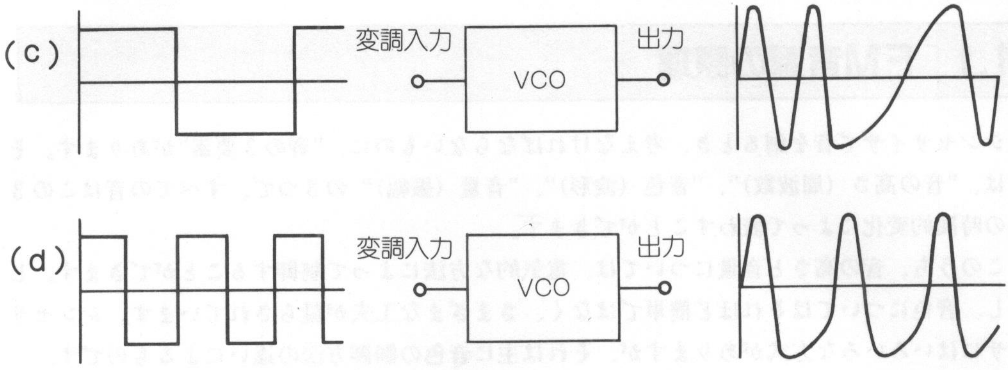


図1-1　周波数変調　その2

以上がFM変調の原理です。ただし、変調入力となるのは方形波に限りません。他の発振器の出力を変調入力として、さらに多彩な波形を発生させることができます。

もうひとつ、発振器の構造について述べておきましょう。アナログシンセサイザでは電気的にサイン波を発生しますが、PCシリーズに搭載されているYM−2203（OPN）はデジタルにサイン波を発生します。具体的には、サイン関数の値をテーブルとして持ち、これを順次読み出してサイン波を発生し、変調入力に応じて読み出す順番を変えることによってさまざまな波形を発生しています。

OPNは、音の高さと音色を定める発振器（オペレータと呼ぶ）の他、音量を定めるエンベロープジェネレータ、従来のPSGと同じ矩形波音源、その他の付属回路によって構成されています。内部構成を次に示します。

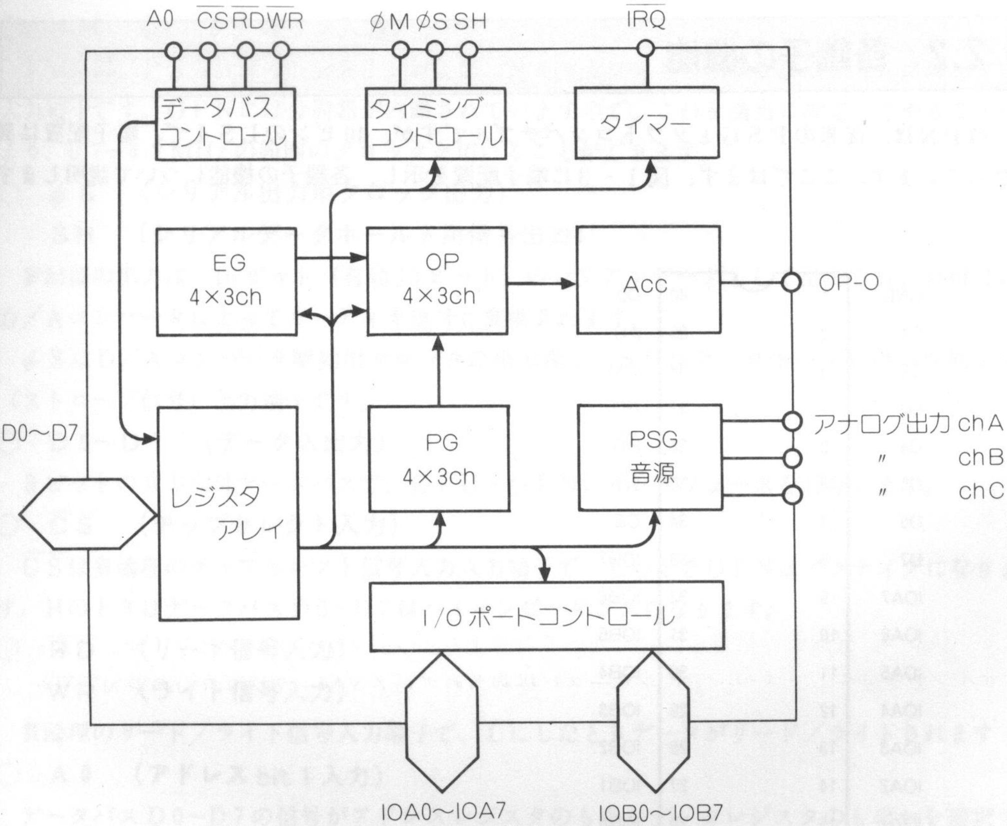
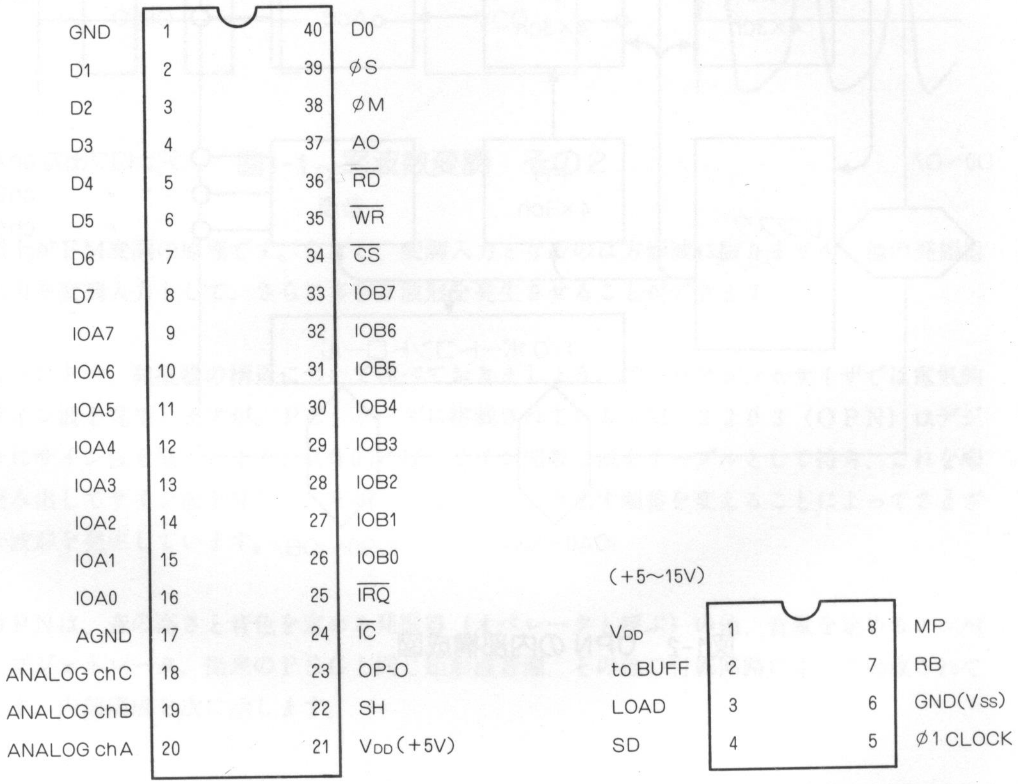


図1-2 OPNの内部構成図

2.2　各端子の機能

ＯＰＮは、従来のＰＳＧとソフトコンパチブルですが、40ピンのＬＳＩで、端子配置は異なっています。ここではまず、図１－３に端子配置を示し、各端子の機能について説明します。



（a） YM-2203　OPN　　　（b） YM-3014　D/Aコンバータ

図1-3　ピン配置

○　φM　(マスタクロック入力)

FM音源、PSG音源の波形発生、および、タイマカウントの基準となるマスタクロックの入力端子です。OPNには分周器が内蔵されていますので、これを適当に設定してやることにより、0.7〜4.2MHzの範囲のクロックを用いることができます。

○　φS　(シリアル出力用クロック出力)

SH　(シリアルデータホールド用信号出力)

FM部の出力は、16ビット(有効13ビット)のシリアルデータとして出力され、外付けのD/Aコンバータによってオーディオ信号に変換されます。

φSはD/Aコンバータ駆動用クロックの出力端子、SHはデータホールド用の周期信号(ストローブ信号)出力端子です。

○　D0〜D7　(データ入出力)

8ビットの双方向性データバスで、CPUとOPNのインタフェースを行ないます。

○　$\overline{\text{CS}}$　(チップセレクト入力)

$\overline{\text{CS}}$は負論理のチップセレクト信号入力入力端子で、LのときOPNはアクティブになります。HのときはデータバスD0〜D7はハイインピーダンスになります。

○　$\overline{\text{RD}}$　(リード信号入力)

$\overline{\text{WR}}$　(ライト信号入力)

負論理のリード/ライト信号入力端子で、Lにしたときデータがリード/ライトされます

○　A0　(アドレスbit 1入力)

データバスD0〜D7の信号がアドレスレジスタのものかデータレジスタのものかを選択する信号です。Lのときアドレスレジスタ、Hのときデータレジスタを選択します。

○　$\overline{\text{IRQ}}$　(インタラプトリクエスト信号出力)

OPNに内蔵されている2系統のタイマのどちらかがセットした時間に達すると、この端子がLになります。2つのタイマのどちらが指定時間に達したのかは、OPNのステータス信号によって判断します。なお、SRシリーズでは、この信号は$\overline{\text{INT3}}$として扱われています

○　$\overline{\text{IC}}$　(クリア)

負論理のリセット信号入力端子で、LにするとOPNのすべてのレジスタの内容がクリアされます。

1.3 レジスタ

1．3．1 レジスタの概要

ＯＰＮは音をつくるＬＳＩですから、周波数、音色、音量、およびそれらの時間的変化などといった情報を扱います。ＯＰＮ内部には、これらの情報を格納するレジスタがあり、レジスタの値を設定することによって音に関するデータをＯＰＮに渡すことができます。 しかし、これらの情報は、６音分で百数十バイトにもなり、ひとつひとつのレジスタに対してＣＰＵのＩ／Ｏアドレスを割り当てると、Ｉ／Ｏアドレス空間が足りなくなってしまいます。

そこでＯＰＮでは、８ビットのアドレスラッチによってＩ／Ｏの対象とするレジスタアドレスを指定し、そのレジスタとデータをやりとりする方法をとっています。この方法によれば、ＣＰＵのＩ／Ｏアドレスはアドレスラッチとデータバッファの２つにのみ割り当てればよいことになります。

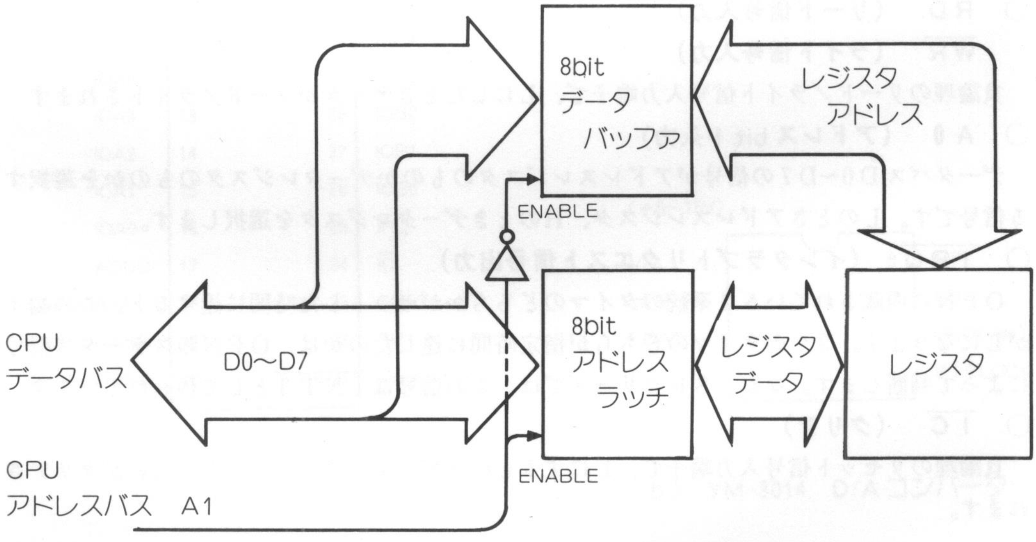


図1-4 アドレスラッチによるアドレシング

具体的なＩ／Ｏアドレスは、ＣＰＵとのインタフェースによって定まり、それぞれ以下のようになっています。

アドレスラッチのＩ／Ｏアドレス　＝　８８Ｈ
データバッファのＩ／Ｏアドレス　＝　８ＡＨ

データ設定の具体例として、ＰＳＧ部のchA（ＢＡＳＩＣではch４にあたる）の音量を０に設定する場合を考えます。

まず、アドレスラッチをセットします。ＰＳＧ部chAの音量設定用レジスタアドレスは０８Ｈですので、ＢＡＳＩＣで、

```
OUT &H88, &H08
```

を実行すれば、アドレスラッチにレジスタアドレス０８Ｈが設定されます。

続いてレジシタデータをセットします。これもＢＡＳＩＣで、

```
OUT &H8A, &H0
```

を実行することにより行なわれます。

このようにＯＰＮにデータを与える場合、「何をどうする」の「何を」にあたるデータ（レジスタアドレス）をＩ／Ｏポート８８Ｈに、「どうする」にあたるデータ（レジスタデータ）をＩ／Ｏポート８Ａに出力します。

また、この操作を簡単に行なうために、ＶＯＩＣＥ　ＲＥＧ命令が用意されています。これを用いると、上と同じ動作を、

```
VOICE REG &H08, &H0
```

と実行することにより行なえます。つまり、ＶＯＩＣＥ　ＲＥＧ命令の第一パラメータはＩ／Ｏポート８８Ｈに、第２パラメータはＩ／Ｏポート８ＡＨにそれぞれ出力されるわけです。

1．3．2　PSG部のレジスタ

PSG部のレジスタは、レジスタアドレス００H～０FHに割り当てられています。各レジスタの意味などについては、〈1．4　PSG部の動作〉を参照してください。

レジスタアドレス					コメント
0 0	Fine Tune				chA 周波数
0 1		Coarse Tune			
0 2	Fine Tune				chB 周波数
0 3		Coarse Tune			
0 4	Fine Tune				chC 周波数
0 5		Coarse Tune			
0 6		Period Controll			ノイズ周波数
0 7	IN/OUT	/Noise	/Tone		I/Oポート/ミキサー
0 8		M	Level		chA レベル
0 9		M	Level		chB レベル
0 A		M	Level		chC レベル
0 B	Fine Tune				エンベロープ周期
0 C	Coarse Tune				
0 D		C	ATT	ALT / HLD	エンベロープ形状
0 E	I/O Port-A				I/Oポートデータ
0 F	I/O Port-B				

図1-5　PSG部レジスタマップ

PSG部のレジスタは、レジスタデータを読み出すことができます。これは、レジスタアドレス0EH，0FHに割り当てられている、OPNの汎用I／Oポートを読む時などに必要な機能です。I／Oポートについては、〈1．6　補助機能〉を参照してください。

たとえば、レジスタアドレス０９Hのデータを読み込む場合、

```
OUT  &H88，&H09
D=INP (&H8A)
```

と実行します。

OPNのステータスを読むには、

```
S=INP (&H88)
```

と実行します。この場合、アドレスラッチにレジスタアドレスをセットする必要はありません。

1．3．3　FM部のレジスタ

FM部のレジスタは、レジスタアドレス21H～B2Hに割り当てられています。各レジスタの意味などについては、〈1．5　FM部の動作〉を参照してください。

ただし、レジスタアドレス21H～27H、2DH～2FHは、むしろOPN全体について制御している部分ですので、〈1．6　補助機能〉で説明します。

レジスタアドレス							コメント
21	TEST						LSIのテストデータ
24	TIMER-A						タイマAのプリセット値
25						TIMER-A	
26	TIMER-B						タイマBのプリセット値
27	Mode		RESET B A		ENABLE B A	LOAD B A	ch3のモード，およびタイマのコントロール
28	Op1	Op2	Op3	Op4		CH	key-on/off
2D	——						分周器のセット
2E	——						⅓，⅙分周の選択
2F	——						½分周にセット

図1-6　FM部レジスタマップ　その1

レジスタアドレス			コメント
30 ～ 3E	Detune	Multiple	
40 ～ 4E	Total Level		×0 ch1 Op1 ×1 ch2 Op1 ×2 ch3 Op1 ×3 dummy ×4 ch1 Op3 ×5 ch2 Op3 ×6 ch3 Op3 ×7 dummy ×8 ch1 Op2 ×9 ch2 Op2 ×A ch3 Op2 ×B dummy ×C ch1 Op4 ×D ch2 Op4 ×E ch3 Op4
50 ～ 5E	Key-Scale	Attack Rate	
60 ～ 6E	Decay Rate		
70 ～ 7E	Sustain Rate		
80 ～ 8E	Sustain Level	Relese Rate	
90 ～ 9E	SSG-EG		
A0 A1 A2	F-Num1		F-Numbers/BLOCK 各々 ch1, ch2, ch3 に対応する
A4 A5 A6	BLOCK	F-Num2	
A8 A9 AA	ch3 F-Num1		効果音モード，音声合成 モードのとき使用
AC AD AE	ch3 BLOCK	ch3 F-Num2	
B0 B1 B2	Feedback	Algorithm	フィードバック/ アルゴリズム

図1-6　FM部レジスタマップ　その2

1.4 ＰＳＧ部の動作

1．4．1 矩形波音源

まず、ＰＳＧ部のブロック図を示します。

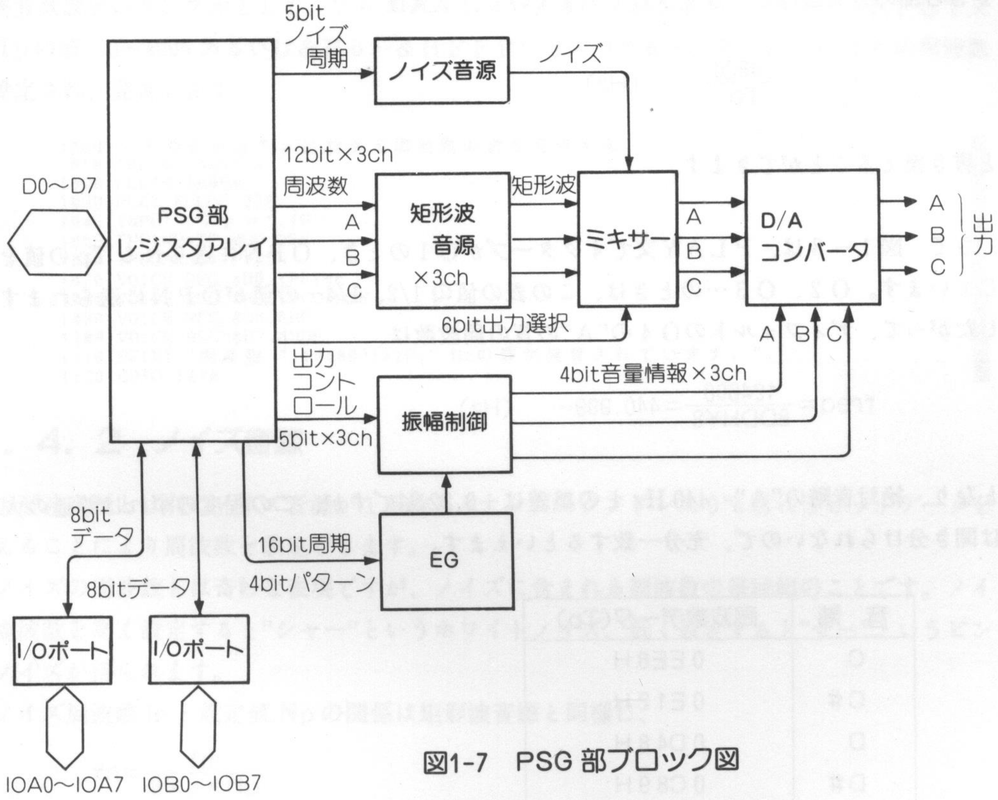


図1-7 PSG部ブロック図

主たる音源は、3系統ある矩形波音源で、各系統ごとに12ビットのピッチ情報で周波数を設定できます。ただし、レジスタのデータ長は8ビットですので、8ビットのFine Tuneと4ビットのCoarse Tuneの2つのレジスタに分けて扱います。出力周波数は、次式により計算できます。

$$freq = \frac{fm}{8 \times n \times Tp}$$

fm ：マスタクロック周波数
n ：内蔵分周器の分周数
Tp ：Coarse Tuneを上位、Fine Tuneを下位とする12ビットデータ

入力クロックとしてはＣＰＵと同じ3.9936MHzが与えられており（ただしボード上で発生）、ＰＳＧ部の分周数は４に設定されていますので、上式は

$$freq = \frac{124800}{Tp} \quad (Hz)$$

と書き換えることができます。

さて、図１－８は、ＰＬＡＹ文でオクターブがＯ１のとき、ＯＰＮに送られるTpの値を示しています。Ｏ２、Ｏ３…のときは、この表の値の1/2、1/4…の値がＯＰＮに送られます。したがって、ディフォルトのＯ４の”A”の音の周波数は、

$$freq = \frac{124800}{8DDH ¥ 8} = 440.989\cdots \quad (Hz)$$

となり、絶対音階の”A”= 440 Hzとの誤差は＋0.22％ですが、この程度のずれは普通の人には聞き分けられないので、充分一致するといえます。

音　階	周波数データ(Tp)
C	0EE8H
C#	0E12H
D	0D48H
D#	0C89H
E	0BD5H
F	0B2BH
F#	0A8AH
G	09F3H
G#	0964H
A	08DDH
A#	085EH
B	07E6H

図1-8　PSG音階データ

この方式では、1/Tpの範囲の誤差は避けられません。誤差はTpが小さくなり周波数が高くなるほど増大し、最高音のO8B音では半音近くにもなりますので、和音を出すときなどは注意してください。

PSG音源で出せる周波数の範囲は、30.5 Hz(Tp = FFFH)～124800 Hz(Tp = 1またはTp = 0)です。ただし、PCシリーズの出力回路には混変調を防ぐためのフィルタが入っていますので、実用になるのは10 kHz(Tp = 12)ぐらいまででしょう。可聴周波数は20～20 kHz程度ですから、これで充分です。

周波数設定のサンプルとして、プログラム1－1を挙げておきます。110行のINPUT文でTpの値（0～4095あるいは&H0～&HFFF）を入力すると、PSG音源chAの周波数が設定され、発音します。

```
1000 ' 入力されたTpに対する周波数の音を発音する
1010 'progl.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 INPUT "Tp = ",TP
1050 TPFINE=TP MOD 256
1060 TPCOARSE=TP¥256
1070 VOICE REG &H0,TPFINE
1080 VOICE REG &H1,TPCOARSE
1090 VOICE REG &H8,&HF
1100 VOICE REG &H7,&HBE
1110 PRINT "周波数 ";124800!/TP;" Hzの音が発音されています。"
1120 GOTO 1040
```

1.4.2 ノイズ音源

矩形波音源とは別にノイズ音源が1系統あり、レジスタアドレス06Hに5ビットデータを与えることにより周波数を設定できます。

ノイズの周波数とは奇妙な表現ですが、ノイズに含まれる周波数の帯域幅のことです。ノイズ周波数を高く設定すると”シャー”というホワイトノイズ、低く設定すると”ザー”というピンクノイズが得られます。

ノイズ周波数fnと設定値Npの関係は矩形波音源と同様に、

$$fn=\frac{fm}{8\times n\times Np}$$

で与えられます。n＝4ですから、

$$fn=\frac{124800}{Np}$$

となります。

Npは5ビットデータですからノイズ周波数の範囲は、4026～124800 Hzということになります。

サンプルとして、プログラム1－2を掲げます。Npの値（0～31　あるいは&H0～&H1F）を入力すると対応する周波数のノイズが出力されます。

```
1000 '入力されたNpに対する周波数のノイズを発生する。
1010 'prog2.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC ,,,256
1040 INPUT "N p = ",NP
1050 IF NP<0 OR NP>31 THEN 1040
1060 IF NP=0 THEN NP=1
1070 VOICE REG &H6,NP
1080 VOICE REG &H8,&HF
1090 VOICE REG &H7,&H87
1100 PRINT "周波数 ";124800!/NP;" Hzのノイズが発生しています。"
1110 GOTO 1040
```

1．4．3 ミキサー

ミキサーは、PSG音源の矩形波とノイズを重ね合わせる機能を持っていますが、各チャンネルの音を出すかどうか、各チャンネルにノイズを混ぜるかどうかを決めるセレクタと考えるとわかりやすいでしょう。レジスタアドレス07Hのbit 0～bit 2が矩形波音源のchA～chCを発音するかどうか、bit 3～bit 5がchA～chCにノイズを混ぜるかどうかのスイッチになっています。負論理ですので、0のときOn，1のときOffとなります。

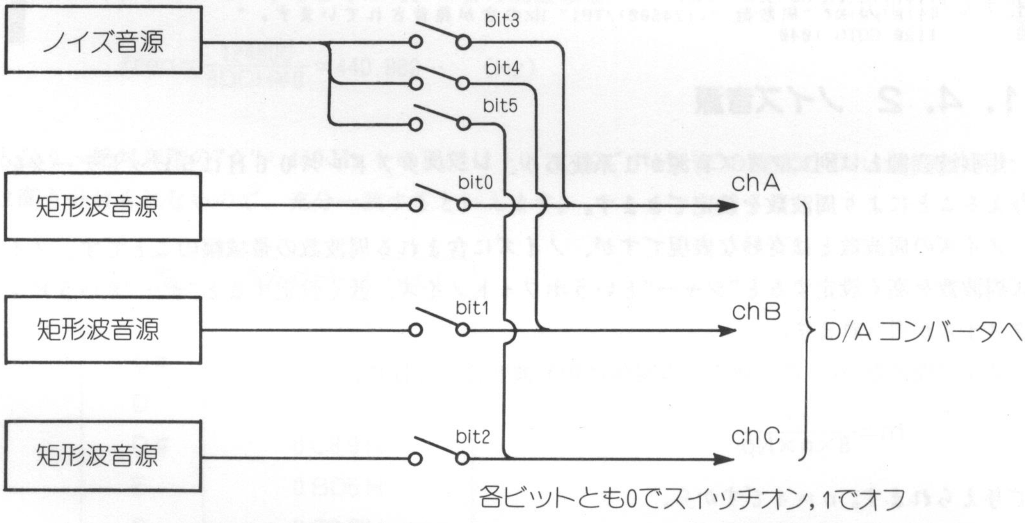


図1-9 ミキサー

1．4．4　振幅コントローラ

音量を決めるボリュームに相当する部分で、レジスタアドレス０８Ｈ～０ＡＨが chA～chC に対応します。

各レジスタアドレスの bit 4 はモード指定スイッチで、０のとき固定振幅モード、１のとき可変振幅モードとなります。

固定振幅モードの場合は、bit 3～bit 0 でレベルを定められる定常音（音量の時間的変化のない音）が出力されます。この場合、設定値をｖ（１～15）とすると出力電圧Ｖ０は、

$$V_0 = 2^{(v-15)/2}$$

となり、人間の聴覚に合わせた指数的特性を持たせてあります。つまり、ｖ＝15のとき出力が最大となり、ｖが１減少するごとに出力電圧は３dB 減少します。

ｖ＝０の場合は、出力は０となります。

可変振幅モードの場合は、次に述べるエンベロープジェネレータにより、音量が時間的に変化する音が出力されます。

1．4．5 エンベロープジェネレータ(EG)

可変振幅モードのとき、音量に時間的変化を与えるものです。

音量変化のパターン（エンベロープ形状）は、レジスタアドレス0DHに与える4ビットのデータによって選択できます。

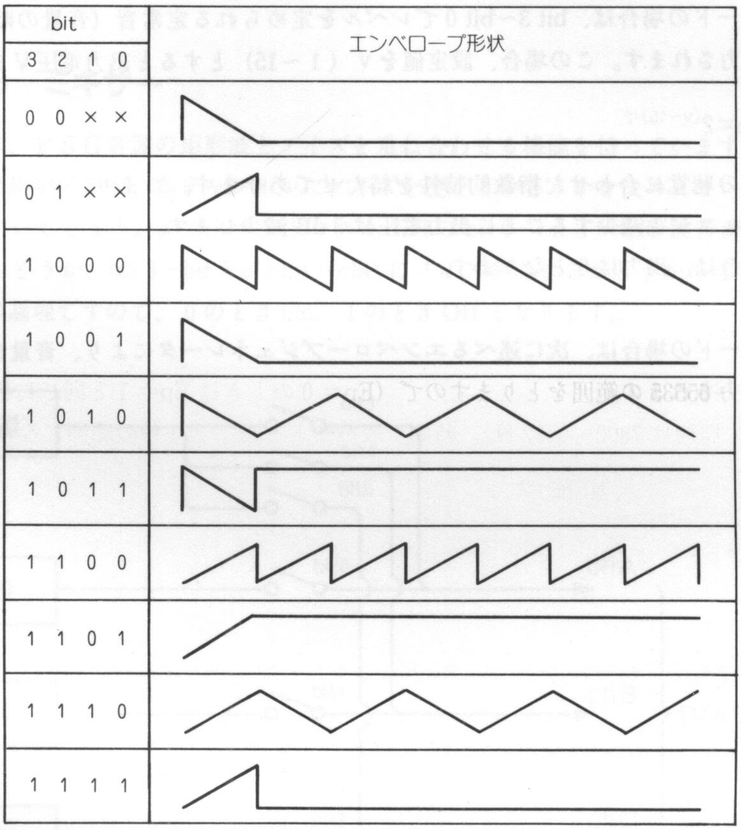


図1-10 PSGエンベロープ形状

音量変化のはやさ（エンベロープ周波数f_E）は、レジスタアドレス０ＢＨに与えるFine Tuneとレジスタアドレス０ＣＨに与えるCoarse Tuneとの計16ビットの情報によって決まります。具体的には、

$$f_E=\frac{fm}{128\times n\times Ep}$$

fm ：マスタクロック周波数
n ：内蔵分周器の分周数
Ep ：Coarse Tuneを上位、Fine Tuneを下位とする16ビットデータ

で定められます。finとnは決定していますので、

$$f_E=\frac{7800}{Ep} \quad (Hz)$$

時間で表わすと、

$$t_E=\frac{Ep}{7800} \quad (秒)$$

となります。

Epの値は１～65535の範囲をとりますので（Ep＝０のときはEp＝１と同じに扱われる）、エンベロープの周期は7800～0.12 Hz、時間で表わすと0.13～8.4秒の範囲になります。

エンベロープによる音量の変化は、エンベロープ形状をセットした時点で始まります。また、EGは１系統しかありませんので、複数のチャンネルを可変振幅モードに設定した場合、すべてのチャンネルが同一の変化をすることになります。

１．４．６　D/Aコンバータ

振幅コントローラからの信号を受けて出力レベルを設定する部分です。

D／Aコンバータ自体は５ビットの分解能を持っていますが、固定振幅モードのときは上位４ビットのみが使われています。可変振幅モードのときは５ビットすべてが使われますが、それでも１ビットあたり1.5 dBに相当しますので、エンベロープ周期を長くとった場合、音量が階段状に変化しているのがわかります。たとえば、

PLAY,,,"T32S0M6500\0C1"

などと実行してみてください。

1.5　FM部の動作

1．5．1　FM部の内部構成

まず、FM部のブロック図を示します。

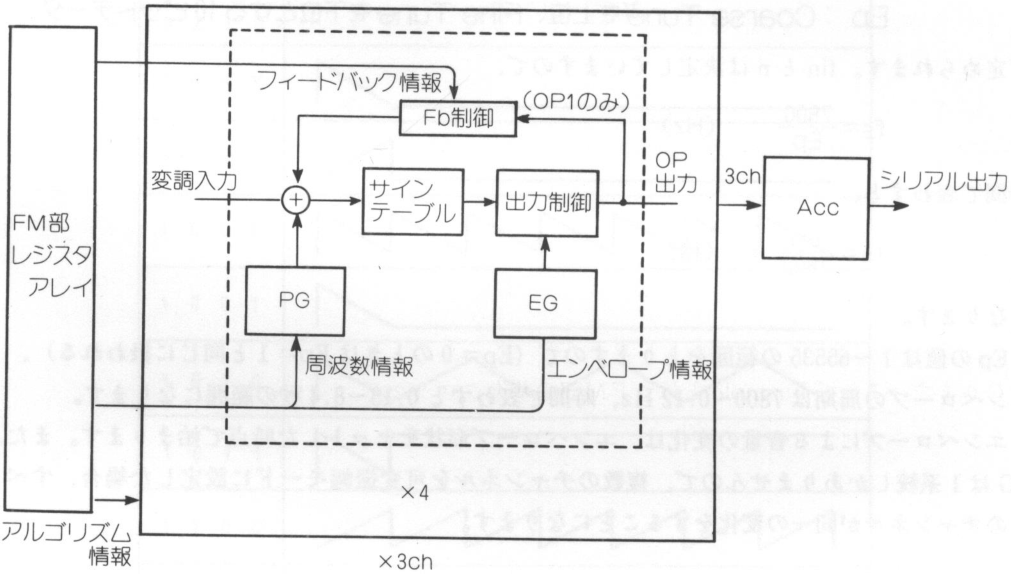


図1-11　FM 部ブロック図

図1－11からわかるように、FM部の基本となるのはオペレータです。3つのチャンネルについてそれぞれ4つ、計12のオペレータがあります。このため、レジスタ数も多くなり、全体の構成がわかりにくいようですが、ひとつのオペレータについて理解できれば、あとは同じものが並んでいるだけです。

オペレータ内部の各ブロックについては、音の3要素である周波数、音色（波形）、音量、およびそれらの時間的変化に対応させて、

周波数を決める	フェイズジェネレータ（PG）
音色（波形）を決める	アルゴリズム（Al）、フィードバック（Fb）
音量とその時間的変化を決める	エンベロープジェネレータ（EG）

と考えるとよいでしょう。

1．5．2 フェイズジェネレータ

○ BLOCK、F-Number

ＰＳＧ部では周波数を定めるのに基準クロックを分周する方法をとっていましたが、この方法では、高い音ほど誤差が大きくなるという欠点がありました。

そこでFM部では、基準クロックを最大分周したときの周波数 Fmin を基準として、発音する周波数 Fmusic がその何倍であるかという情報 F-Number を与えて周波数を定める方法をとっています。ただし、発音周波数帯域全体をカバーするには 20 ビット程度の情報が必要ですので、オクターブを表わす 3 ビットの情報 BLOCK とオクターブ内の音程を表わす 11 ビットの F-Number に分けて扱っています。

具体的に式で表わすと

$$\text{F-Number} = \frac{\text{Fmusic}}{\text{Fsample} \times 2^{\text{BLOCK}-21}}$$

あるいは逆に

$$\text{Fmusic} = \text{F-Number} \times \text{Fsample} \times 2^{\text{BLOCK}-21} \quad (\text{Hz})$$

となります。ここで Fsample とは、ＦＭ部の基準クロックで、

$$\text{Fsample} = \frac{\text{fm}}{12 \times n}$$

で表わされます。Fm＝3.9936MHz，n＝6ですから、

$$\text{F-Number} = \frac{3 \times \text{Fmusic}}{166400 \times 2^{\text{BLOCK}-21}}$$

$$\text{Fmusic} = \text{F-Number} \times \frac{166400}{3} \times 2^{\text{BLOCK}-21} \quad (\text{Hz})$$

となります。

次の表は、各音程に対する F-Number の値を示したものです。この表から、たとえばオクターブ 4 の“A”音の周波数を計算すると、

$$\text{Fmusic} = 410\text{H} \times \frac{166400}{3} \times 2^{(4-21)} = 440.104\cdots \quad (\text{Hz})$$

となります。

N	Oct		BLOCK	F-Num	Fmusic (Hz)	Fdelta (Hz) Dt=0	Dt=1	Dt=2	Dt=3	Weight of Rate KS=0	KS=1	KS=2	KS=3
1	1	C#	0	028FH	35	0.000	0.000	0.053	0.106	0	0	0	0
2	1	D	0	02B6H	37	0.000	0.000	0.053	0.106	0	0	0	0
3	1	D#	0	02DFH	39	0.000	0.000	0.053	0.106	0	0	0	0
4	1	E	0	030BH	41	0.000	0.000	0.053	0.106	0	0	0	0
5	1	F	0	0339H	44	0.000	0.000	0.053	0.106	0	0	0	0
6	1	F#	0	036AH	46	0.000	0.000	0.053	0.106	0	0	0	0
7	1	G	0	039EH	49	0.000	0.000	0.053	0.106	0	0	0	1
8	1	G#	0	03D5H	52	0.000	0.000	0.053	0.106	0	0	0	1
9	1	A	0	0410H	55	0.000	0.000	0.053	0.106	0	0	1	2
10	1	A#	0	044EH	58	0.000	0.000	0.053	0.106	0	0	1	2
11	1	B	0	048FH	62	0.000	0.000	0.053	0.106	0	0	1	3
12	2	C	1	026AH	65	0.000	0.053	0.106	0.106	0	1	2	4
13	2	C#	1	028FH	69	0.000	0.053	0.106	0.106	0	1	2	4
14	2	D	1	02B6H	73	0.000	0.053	0.106	0.106	0	1	2	4
15	2	D#	1	02DFH	78	0.000	0.053	0.106	0.106	0	1	2	4
16	2	E	1	030BH	82	0.000	0.053	0.106	0.106	0	1	2	4
17	2	F	1	0339H	87	0.000	0.053	0.106	0.106	0	1	2	4
18	2	F#	1	036AH	92	0.000	0.053	0.106	0.106	0	1	2	4
19	2	G	1	039EH	98	0.000	0.053	0.106	0.159	0	1	2	5
20	2	G#	1	03D5H	104	0.000	0.053	0.106	0.159	0	1	2	5
21	2	A	1	0410H	110	0.000	0.053	0.106	0.159	0	1	3	6
22	2	A#	1	044EH	117	0.000	0.053	0.106	0.159	0	1	3	6
23	2	B	1	048FH	123	0.000	0.053	0.106	0.159	0	1	3	7
24	3	C	2	026AH	131	0.000	0.053	0.106	0.212	1	2	4	8
25	3	C#	2	028FH	139	0.000	0.053	0.106	0.212	1	2	4	8
26	3	D	2	02B6H	147	0.000	0.053	0.106	0.212	1	2	4	8
27	3	D#	2	02DFH	156	0.000	0.053	0.106	0.212	1	2	4	8
28	3	E	2	030BH	165	0.000	0.053	0.106	0.212	1	2	4	8
29	3	F	2	0339H	175	0.000	0.053	0.106	0.212	1	2	4	8
30	3	F#	2	036AH	185	0.000	0.053	0.106	0.212	1	2	4	8
31	3	G	2	039EH	196	0.000	0.053	0.159	0.212	1	2	4	9
32	3	G#	2	03D5H	208	0.000	0.053	0.159	0.212	1	2	4	9
33	3	A	2	0410H	220	0.000	0.053	0.159	0.212	1	2	5	10
34	3	A#	2	044EH	233	0.000	0.053	0.159	0.212	1	2	5	10
35	3	B	2	048FH	247	0.000	0.053	0.159	0.264	1	2	5	11
36	4	C	3	026AH	262	0.000	0.106	0.212	0.264	1	3	6	12
37	4	C#	3	028FH	277	0.000	0.106	0.212	0.264	1	3	6	12
38	4	D	3	02B6H	294	0.000	0.106	0.212	0.264	1	3	6	12
39	4	D#	3	02DFH	311	0.000	0.106	0.212	0.264	1	3	6	12
40	4	E	3	030BH	330	0.000	0.106	0.212	0.264	1	3	6	12
41	4	F	3	0339H	349	0.000	0.106	0.212	0.264	1	3	6	12
42	4	F#	3	036AH	370	0.000	0.106	0.212	0.264	1	3	6	12
43	4	G	3	039EH	392	0.000	0.106	0.212	0.317	1	3	6	13
44	4	G#	3	03D5H	415	0.000	0.106	0.212	0.317	1	3	6	13
45	4	A	3	0410H	440	0.000	0.106	0.212	0.317	1	3	7	14
46	4	A#	3	044EH	466	0.000	0.106	0.212	0.317	1	3	7	14
47	4	B	3	048FH	494	0.000	0.106	0.264	0.370	1	3	7	15
48	5	C	4	026AH	523	0.000	0.106	0.264	0.423	2	4	8	16
49	5	C#	4	028FH	554	0.000	0.106	0.264	0.423	2	4	8	16
50	5	D	4	02B6H	587	0.000	0.106	0.264	0.423	2	4	8	16
51	5	D#	4	02DFH	622	0.000	0.106	0.264	0.423	2	4	8	16
52	5	E	4	030BH	659	0.000	0.106	0.264	0.423	2	4	8	16
53	5	F	4	0339H	698	0.000	0.106	0.264	0.423	2	4	8	16
54	5	F#	4	036AH	740	0.000	0.106	0.264	0.423	2	4	8	16
55	5	G	4	039EH	784	0.000	0.159	0.317	0.423	2	4	8	17
56	5	G#	4	03D5H	830	0.000	0.159	0.317	0.423	2	4	8	17
57	5	A	4	0410H	880	0.000	0.159	0.317	0.476	2	4	9	18
58	5	A#	4	044EH	933	0.000	0.159	0.317	0.476	2	4	9	18
59	5	B	4	048FH	988	0.000	0.159	0.370	0.529	2	4	9	19

図1-12　FM部周波数テーブル　その1

N	Oct		BLOCK	F-Num	Fmusic (Hz)	Fdelta (Hz)				Weight of Rate			
						Dt=0	Dt=1	Dt=2	Dt=3	KS=0	KS=1	KS=2	KS=3
60	6	C	5	026AH	1046	0.000	0.212	0.423	0.582	2	5	10	20
61	6	C#	5	028FH	1109	0.000	0.212	0.423	0.582	2	5	10	20
62	6	D	5	02B6H	1175	0.000	0.212	0.423	0.582	2	5	10	20
63	6	D#	5	02DFH	1244	0.000	0.212	0.423	0.582	2	5	10	20
64	6	E	5	030BH	1319	0.000	0.212	0.423	0.582	2	5	10	20
65	6	F	5	0339H	1396	0.000	0.212	0.423	0.582	2	5	10	20
66	6	F#	5	036AH	1479	0.000	0.212	0.423	0.582	2	5	10	20
67	6	G	5	039EH	1567	0.000	0.212	0.423	0.635	2	5	10	21
68	6	G#	5	03D5H	1661	0.000	0.212	0.423	0.635	2	5	10	21
69	6	A	5	0410H	1760	0.000	0.212	0.476	0.688	2	5	11	22
70	6	A#	5	044EH	1865	0.000	0.212	0.476	0.688	2	5	11	22
71	6	B	5	048FH	1975	0.000	0.264	0.529	0.741	2	5	11	23
72	7	C	6	026AH	2092	0.000	0.264	0.582	0.846	3	6	12	24
73	7	C#	6	028FH	2217	0.000	0.264	0.582	0.846	3	6	12	24
74	7	D	6	02B6H	2349	0.000	0.264	0.582	0.846	3	6	12	24
75	7	D#	6	02DFH	2488	0.000	0.264	0.582	0.846	3	6	12	24
76	7	E	6	030BH	2637	0.000	0.264	0.582	0.846	3	6	12	24
77	7	F	6	0339H	2793	0.000	0.264	0.582	0.846	3	6	12	24
78	7	F#	6	036AH	2959	0.000	0.264	0.582	0.846	3	6	12	24
79	7	G	6	039EH	3135	0.000	0.317	0.635	0.899	3	6	12	25
80	7	G#	6	03D5H	3321	0.000	0.317	0.635	0.899	3	6	12	25
81	7	A	6	0410H	3521	0.000	0.317	0.688	1.005	3	6	13	26
82	7	A#	6	044EH	3731	0.000	0.317	0.688	1.005	3	6	13	26
83	7	B	6	048FH	3951	0.000	0.370	0.741	1.058	3	6	13	27
84	8	C	7	026AH	4184	0.000	0.423	0.846	1.164	3	7	14	28
85	8	C#	7	028FH	4435	0.000	0.423	0.846	1.164	3	7	14	28
86	8	D	7	02B6H	4699	0.000	0.423	0.846	1.164	3	7	14	28
87	8	D#	7	02DFH	4977	0.000	0.423	0.846	1.164	3	7	14	28
88	8	E	7	030BH	5274	0.000	0.423	0.846	1.164	3	7	14	28
89	8	F	7	0339H	5586	0.000	0.423	0.846	1.164	3	7	14	28
90	8	F#	7	036AH	5918	0.000	0.423	0.846	1.164	3	7	14	28
91	8	G	7	039EH	6270	0.000	0.423	0.846	1.164	3	7	14	29
92	8	G#	7	03D5H	6642	0.000	0.423	0.846	1.164	3	7	14	29
93	8	A	7	0410H	7042	0.000	0.423	0.846	1.164	3	7	15	30
94	8	A#	7	044EH	7461	0.000	0.423	0.846	1.164	3	7	15	30
95	8	B	7	048FH	7902	0.000	0.423	0.846	1.164	3	7	15	31

・図1-12　FM部周波数テーブル　その2

さて、BLOCK とF-Number によって表わされる周波数データは14ビットですが、OPNのレジスタデータ長は8ビットですので、F-Num 1とBLOCK／F-Num 2との2つのレジスタに分けて扱うことになります。

F-Num 1、すなわちF-Numberの下位8ビットは、レジスタアドレスA0H～A2Hがそれぞれch 1～ch 3に対応して割り当てられています。BLOCK／F-Num 2は、レジスタアドレスA4H～A6Hに割り当てられ、それぞれbit 5～bit 3がBLOCK、bit 2～bit 0がF-Num 2すなわちF-Numberの上位3ビットに対応します。

これらのレジスタをセットする際には、必ずBLOCK／F-Num 2を先にセットするようにしてください。というのは、OPNは、F-Num 1がセットされたとき同時にBLOCK／F-Num 2を読み込むので、F-Num 1を先にセットすると以前のBLOCK／F-Num 2の値が読み込まれてしまうからです。

次に示すプログラムは、ch 1 にオクターブ4の”A”音を設定し発音するサンプルです。

```
1000 ' ＦＭ音源でオクターブ４の”Ａ”音を発する。
1010 'prog3.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 F.NUM=&H410
1050 F.NUM1=F.NUM MOD 256
1060 F.NUM2=F.NUM¥256
1070 BLOCK=4-1
1080 BL.F2=BLOCK*8+F.NUM2
1090 VOICE REG &HA4,BL.F2
1100 VOICE REG &HA0,F.NUM1
1110 VOICE REG &H28,&HF0
1120 PRINT "ＦＭ音源で周波数 440 Hzの音を発音しています。"
1130 FOR I=1 TO 5000:NEXT
1140 END
```

○ Detune

ＯＰＮで厚みのある音をつくる場合、複数のオペレータの出力を混ぜ合わせるのですが、アナログ型のシンセサイザと違い各オペレータの位相がまったく揃ってしまうため、そのままでは厚みがでません。たとえば、２つのオペレータのF-Numberを等しく設定して発音すると、単に音量が倍になるだけで音色は少しも変化しません。

そこで、各オペレータの周波数にわずかなずれを与えるDetune機能があります。チャンネル毎に与えられたBLOCK／F-Numberは、各オペレータに渡される際に、オペレータ毎に指定されたDetuneの値から計算したオフセットを加算して渡されます。（さらに、後述のMultipleも関係するのですが、ここでは省略します。）

Detuneの値に対する出力周波数の変化は、単純な式では表わせません。図１−12のFdelta欄に示してありますので、音創りの際に参考にしてください。

○ Multiple

Multipleは、Detune操作を受けたF-Numberに、指定された倍率を乗ずるはたらきを持ちます。つまり、実際に発音される周波数は、F-NumberとDetuneによって決まる値（図１−12参照）にMultipleの倍率をかけたものになります。

Multipleは、レジスタアドレス３０Ｈ〜３ＥＨの下位４ビットに割り当てられており、設定値が１〜15のとき倍率は１倍〜15倍となり、０のとき0.5倍になります。

1．5．3　アルゴリズムとフィードバック

フェイズジェネレータから各オペレータに送られる情報は、単に周波数を指定するだけですので、そのままでは味気ないサインウェーブが発音されます。この周波数情報と変調入力を重ね合わて波形を歪ませせることによってさまざまな音色を得る、というのがFM音源の原理です。（ここでは周波数と言っていますが、実際には変調入力は位相情報と重合わせられます。つまり、厳密にはFM変調ではなくPM変調なのですが、両者は基本的に同じものですので、特に支障はありません。）

変調入力としては、他のオペレータの出力が接続されます。

どのオペレータの出力が接続されるかを決めるのが、アルゴリズムです。OPNでは、図1-14に示す8種類のアルゴリズムを選択することができます。

また、各チャンネルの4つのオペレータのうち、第一オペレータについては、自分自身の出力を変調入力とすることができます。これをフィードバックと呼び、その程度を0～7の値で設定することができます。設定値と変調度の関係を次に示します。

bit5～bit3の値	0	1	2	3	4	5	6	7
変調度	OFF	π/16	π/8	π/4	π/2	π	2π	4π

図1-13　フィードバック量

アルゴリズム／フィードバックは、レジスタアドレスB0H～B2Hがそれぞれch 1～ch 3に対応して割り当てられており、bit 5～bit 3にフィードバックを、bit 2～bit 0にアルゴリズムをセットします。

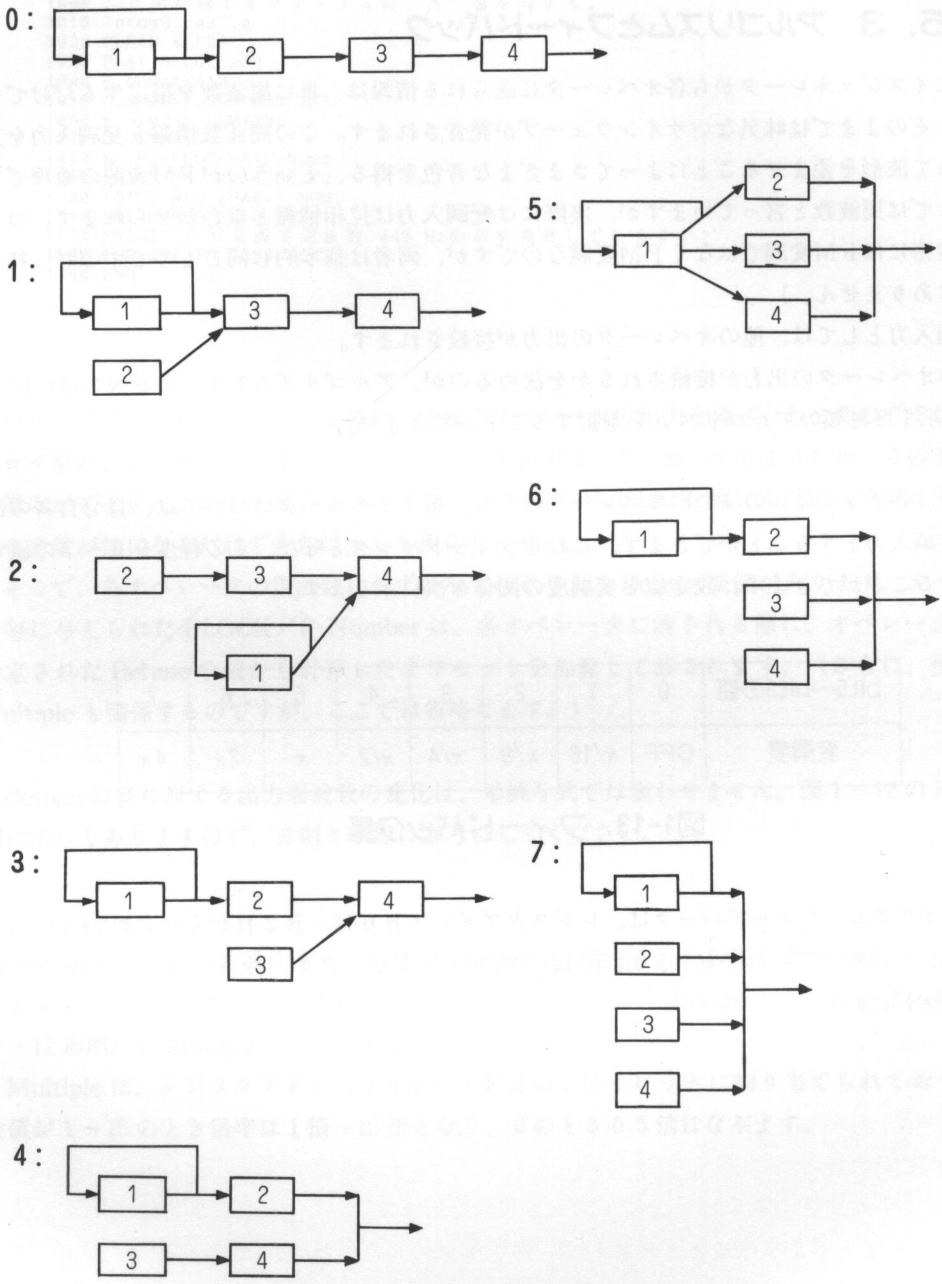


図1-14　アルゴリズム

1. 5. 4 エンベロープジェネレータ

フェイズジェネレータからの情報とアルゴリズム／フィードバックにしたがってサインテーブルから読み出された波形は、エンベロープジェネレータの出力とかけあわせることによってレベルづけされ、つまり音量を決められて、オペレータの出力となります。

一般的に、エンベロープは次に示すような変化をします。

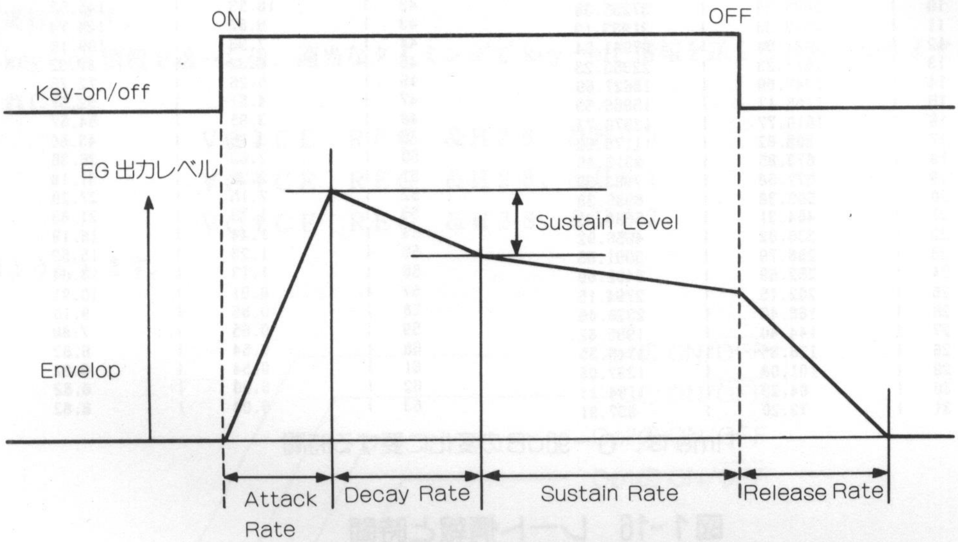


図1-15 エンベロープ変化

アタックレート（AR）は、鍵盤を押すことに相当するkey－on情報が与えられてから、レベルが最大値に達するまでの時間を決める情報です。

ディケイレート（DR）は、レベルが最大値に達してからサスティンレベルに達するまでの時間を決める情報です。

サスティンレート（SR）は、レベルがサスティンレベルに達してから0になるまでの時間を決める情報です。

リリースレート（RR）は、鍵盤を離すことに相当するkey－off情報が与えられてからレベルが0になるまでの時間を決める情報です。

以上4つのレート情報の設定値と時間の関係は、クロック周波数と分周数によって決まりますが、ＳＲシリーズでは、次に示すようになっています。ただし、図中のRateとは、

（レート情報の設定値）×２＋（キースケールによる重みづけ）　を表します。キースケールによる重みづけは、図1-12 Weight欄に示してありますので、参照してください。

Rate	Attack Time (ms)	Decay / Release / Sustain Time (ms)	Rate	Attack Time (ms)	Decay / Release / Sustain Time (ms)
0	---------	---------	32	63.17	873.17
1	---------	---------	33	50.54	698.54
2	---------	---------	34	42.12	582.12
3	---------	---------	35	36.10	498.96
4	8086.15	111766.15	36	31.59	436.59
5	8086.15	111766.15	37	25.27	349.27
6	5390.77	74510.78	38	21.06	291.06
7	5390.77	74510.78	39	18.05	249.48
8	4043.08	55883.08	40	15.79	218.29
9	3234.47	44706.46	41	12.63	174.63
10	2695.38	37255.38	42	10.53	145.53
11	2310.33	31933.19	43	9.02	124.74
12	2021.54	27941.54	44	7.90	109.15
13	1617.23	22353.23	45	6.32	87.32
14	1347.69	18627.69	46	5.26	72.76
15	1155.17	15966.59	47	4.51	62.37
16	1010.77	13970.77	48	3.95	54.57
17	808.62	11176.62	49	3.16	43.66
18	673.85	9313.85	50	2.63	36.38
19	577.58	7983.30	51	2.26	31.18
20	505.38	6985.38	52	2.16	27.29
21	404.31	5588.31	53	1.73	21.83
22	336.92	4656.92	54	1.44	18.19
23	288.79	3991.65	55	1.23	15.59
24	252.69	3492.69	56	1.13	13.64
25	202.15	2794.15	57	0.91	10.91
26	168.46	2328.46	58	0.85	9.10
27	144.40	1995.82	59	0.65	7.80
28	126.35	1746.35	60	0.54	6.82
29	101.08	1397.08	61	0.54	6.82
30	84.23	1164.23	62	0.00	6.82
31	72.20	997.91	63	0.00	6.82

Time は、0～96dBの変化に要する時間

図1-16　レート情報と時間

サスティンレベル（SL）は、ディケイとサスティンの変換点を決める情報で、変換点におけるレベルをデータとして与えます。

トータルレベル（TL）は、エンベロープジェネレータの出力レベルを決める情報で、これにより全体の音量制御を行ないます。

以上２つのレベル情報は、どちらも最大値に対する減衰量（dB）で表現することに注意してください。各ビットの重みづけは次に示すとおりです。

(a)
サスティンレベル

80～8EH	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
減衰量(dB)	24	12	6	3	／	／	／	／

(b)
トータルレベル

40～4EH	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
減衰量(dB)	／	48	24	12	6	3	1.5	0.75

図1-17　レベル情報の重みづけ

エンベロープジェネレータを制御するには、key−on/key−off 情報を与えなければなりません。この情報には、レジスタアドレス２８Hが割り当てられています。図１−18 に示すように、bit 1〜bit 0でチャンネルを指定し、bit 7〜bit 4で各オペレータのキー情報を与えます。たとえば、ch 1〜ch 3のすべてのエンベロープジェネレータに key−on 情報を与えるには、

```
VOICE REG &H28, &HF0
VOICE REG &H28, &HF1
VOICE REG &H28, &HF2
```

と実行します。

key−on 情報を送った後、適当なタイミングで key−off 情報を送らなければなりません。それには、

```
VOICE REG &H28, &H00
VOICE REG &H28, &H01
VOICE REG &H28, &H02
```

のようにします。

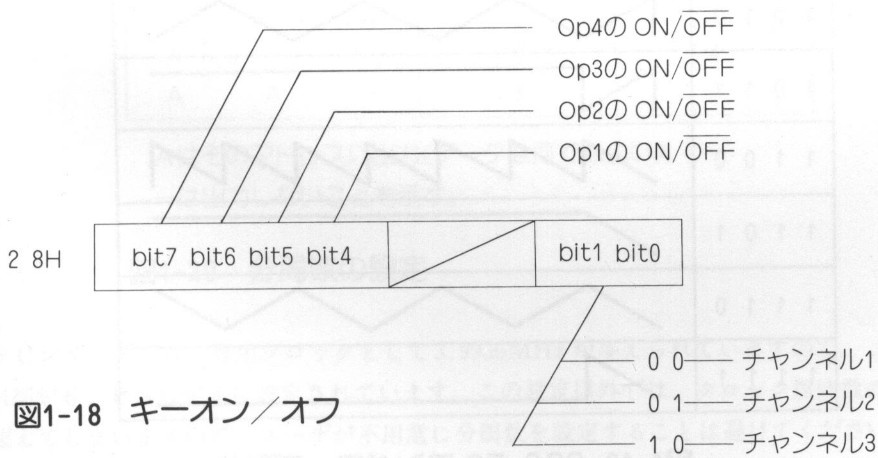


図1-18 キーオン／オフ

楽器音のなかには、高い音ほどエンベロープの変化が速いものが多くあります。このような音を効率よくつくるために、キースケール（KS）という機能が用意されています。 キースケールは、レジスタアドレス５０H〜５EHの bit 7〜bit 6 に割り当てられていて、設定値が大きいほど高音でのエンベロープ変化が速くなります。

このようにOPNのエンベロープジェネレータは多くの機能を持っていますが、欠点として、LFO（Low Frequency Oscilator 変調用の超低周波発振器でワウやトレモロ効果を出すの

に用いる）を内蔵していないことが挙げられます。

その代わりとして、SSG－EGという機能があります。これは、前述のＰＳＧ部に内蔵されているエンベロープジェネレータと同様の形状のエンベロープを発生するものです。SSG－EGには、レジスタアドレス９０Ｈ～９ＥＨの下位４ビットが割り当てられています。bit 4 はSSG－EGのスイッチで１のときSSG－EGがアクティブになり、そのときエンベロープ形状はbit 3～bit 0に対応して、下図のように定まります。エンベロープ周期はディケイレートとサスティンレートによって決まります。リリースレートは通常と同じくkey－off後の減衰を表わし、アタックレートは１ＦＨに設定するのが標準です。

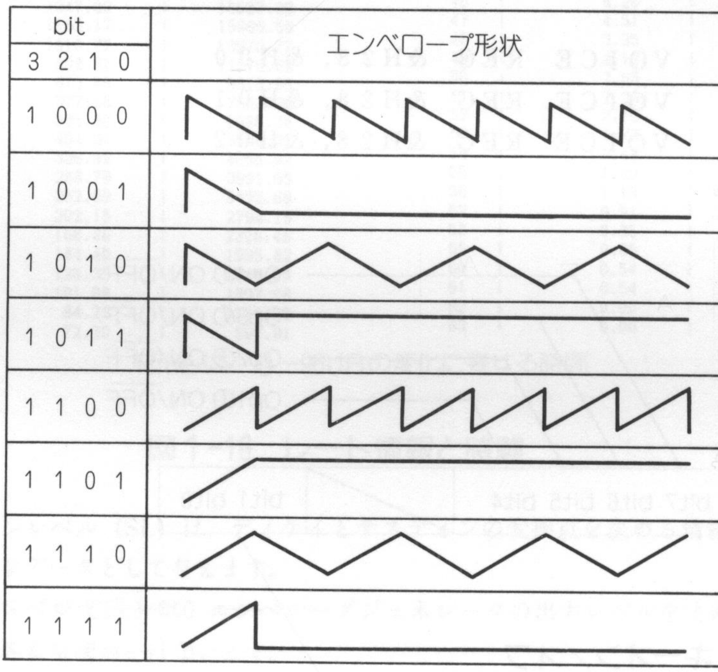


図1-19 SSG-EGエンベロープ形状

1.6　補助機能

1．6．1　分周器

ＯＰＮが動作するためには、マスタクロックが必要ですが、このクロックを得るために外付け回路を必要としないよう、プリセッタブルな分周器を内蔵しています。この分周器を適当に設定してやることによって0.7~4.2MHzのクロックをマスタクロックとして使用できるようになるのです。

分周数は、ＯＰＮをリセットした直後はＦＭ部に６、ＰＳＧ部に４が設定されています。分周器にはレジスタアドレス２ＤＨ～２ＦＨが割り当てられており、下図に示すレジスタアドレスをセットすることにより(レジスタデータは不要)、分周数が設定されます。

2DH	2EH	2FH	FM部分周数	PSG部分周数
—	—	A	2	1
A	—	—	6	4
A	A	—	3	2

Aはそのアドレスに出力(データは何でもよい),
—は出力しないことを示す。

図1-20　分周数の設定

ＰＣシリーズでは、専用クロックとして3.9936MHzが与えられていますので、分周数は、ＦＭ部が６、ＰＳＧが４に設定されています。この設定以外では、クロック周波数の許容範囲を超えてしまいますので、ユーザが不用意に分周数を設定することは避けてください。

１．６．２ タイマ

ＯＰＮには、Ａ、Ｂ２系統のタイマが用意されています。どちらも汎用のインタバルタイマですが、ＰＣシリーズでは、タイマＢをＰＬＡＹ文のテンポ計算用に使用しています。

タイマのコントロールは主としてレジスタアドレス２７Ｈのbit 5~bit 0で行ないます。各ビットの意味は次のとおりです。

bit 0　タイマＡの始動／停止を制御します。このビットをＨにすると、プリセット値をロードしてタイマＡを始動します。プリセット値は、レジスタアドレス２４Ｈを上位、２５Ｈのbit 1~bit 0を下位とする10ビットデータです。このビットをＬにするとタイマＡは停止します。

bit 1　タイマＢの始動／停止を制御します。タイマＢに対し、bit 0と同様の動作をします。プリセット値はレジスタアドレス２６Ｈの８ビットデータです。

bit 2　タイマＡのＩＲＱイネーブルフラグです。このビットをＨにすると、タイマＡのオーバーフローに際して割込み信号を発生します。（具体的には、$\overline{\text{ＩＲＱ}}$端子がＬになる。）　このとき、ステータス信号を読むと、bit 0タイマＡフラグがセットされています。

bit 3　タイマＢのＩＲＱイネーブルフラグです。タイマＢ割込みが発生すると、ステータス信号のbit 1がセットされます。

bit 4　このビットをＨにするとタイマＡ割込み信号を解除します。

bit 5　このビットをＨにするとタイマＢ割込み信号を解除します。

インタバルタイムは、プリセット値とマスタクロック、およびFM部の分周数によって決まります。タイマＡ、ＢのインタバルタイムをそれぞれT0va、T0vb、プリセット値をNa、Nbとすると、その関係は次の式で表わされます。

$$T0_{va}=\frac{12\times n\times(1024-Na)}{fm}$$

$$T0_{vb}=\frac{192\times n\times(256-Nb)}{fm}$$

n　：FM部の分周数
fm　：マスタクロック

ＰＣシリーズでは、fm = 3.9936 MHz、n = 6ですから、

$$T0_{va} = (1024 - Na) \times \frac{3}{166400} \quad (秒)$$

$$T0_{vb} = (256 - Nb) \times \frac{3}{10400} \quad (秒)$$

となります。また、Naは11ビット、Nbは8ビットですから、T0va、T0vbの範囲はそれぞれ0.018～18.5(ms)、0.288～73.8(ms)となります。

1．6．3　I／Oポート

ＯＰＮには、Ａ、Ｂ２系統のＩ／Ｏポートが、従来のＰＳＧとの互換性を保つためにつけられています。各Ｉ／Ｏポートは、レジスタアドレス０ＥＨ，０ＦＨのデータとして入出力を行ないます。

ＰＣシリーズでは、ジョイスティック用として使用しています。

1.7 効果音/音声合成モード

1．7．1 モードの選択

ＦＭ部のch３については、拡張機能として、効果音モード(ＳＥ)、音声合成モード(ＣＳＭ) モードを選択することができます。これに対し、これまで述べてきた楽音を合成するモードを、ノーマルモードと呼びます。ch１、ch２ではノーマルモードのみ選択できます。

ch３のモード選択は、レジスタアドレス２７Ｈ bit７〜bit６によって行ないます。

bit7	bit6	モード
0	0	ノーマルモード
0	1	効果音モード
1	0	音声合成モード

図1-21
モードの選択

1．7．2 効果音モード

一般に楽器音は、倍音成分が基音の整数倍の周波数になっていることが多いため、ノーマルモードでは各オペレータの周波数関係はMultipleによって簡単な比になるように設定されます。しかし、楽器音以外の音、つまり効果音の場合は、非整数次倍音を含むことが多く、ノーマルモードでは合成できません。そこで、効果音モードでは、４つのオペレータの周波数を独立に設定することができるようになっています。このとき、F-Numberを設定するレジスタアドレスは次のとおりです。

アドレス	Op1	Op2	Op3	Op4
F-Num1	A9H	AAH	A8H	A2H
F-Num2/BLOCK	ADH	AEH	ACH	A6H

図1-22　ch3 の F-Number アドレス

1．7．3　音声合成モード

音声合成モードを選択すると、複合正弦波（ＣＳＭ）による音声合成が可能です。とはいっても、そのためにはソフトウェアも、音声データも必要になりますので、すぐに音声を出すというわけにはいきません。ここではさしあたり、音声合成モードの機能について述べることにします。

複合正弦波による音声合成はＮＴＴで開発された方法で、数個の正弦波発振器とＶＣＡという、比較的簡単なハードウェアにより実現可能なものです。このハードウェアは、どれもＦＭ音源で使用するものでＯＰＮにもともと備わっているものです。

音声合成モードを選択すると、効果音モードの場合と同様に、ch３の４つのオペレータの周波数が独立に設定できるようになります。レジスタアドレスは、図１－22のとおりです。このとき、エンベロープジェネレータのトリガは、タイマＡによって一定周期で繰り返し発生し、トータルレベルを変更した場合は次のトリガによってレベルの変更が行なわれます。

[illegible]

1.7.2 効果音モード

一般に楽器では、倍音成分が基音の整数倍の周波数になっていることが多いため、ノーマルモードでは各オペレータの周波数関係はMultipleによって簡単な比になるように設定されます。しかし、[illegible]

アドレス	Op1	Op2	Op3	Op4
F-Num1	A9H	AAH	A8H	A2H
F-Num2/BLOCK	ADH	AEH	ACH	A6H

図1-22 ch3のF-Numberアドレス

第2章

音創りの基本

2.1 音とはなにか

FM音源の話をする前に、音とはどういうものか、どういう基準で計ることができるか、どうすれば簡潔に表現できるか、というような音についての基本的な話をしておきます。このような知識があると、音のイメージを的確に表わすことができ、音創りの大きな助けになるからです。

結論から述べましょう。音は、音量、音程、音色という3つの要素に分解されます。音を電気信号に変えてオシロスコープで波形を見ると、視覚的に理解できるでしょう。音量は波の振幅であり、音程は波の周波数、音色は波形に対応しています。このうち、音量、音程はどちらも単純な物理量で、何ホン、何デシベル、何Hzとはっきりした数値で表わせます。しかし、音色はそう単純ではありません。波の形や音色の感じ、そして両者の関係を数値で明確に表わすことはきわめて困難です。

音量や音程は同じ楽器で変えることができますが、音色はおおむね一定です。音が大きくても小さくても、ドの音でもレの音でも、ピアノの音はピアノの音に聞こえます。

いま単に”音”といいましたが、これはつまり”音色”の意味ですね。これを見てもわかるように音色は音の3要素のうちでも特に重要な意味を持っているのです。FM音源についても、音創りということは、ほとんど音色創りを意味しています。

さてそれでは、音色というものについて、詳しく説明しておきましょう。音色を理解するには、まず倍音に対する理解が必要です。楽器音をはじめ、わたしたちが耳にする音のほとんどは倍音を含んでいます。たとえば、ピアノでA3の鍵盤を叩いてみると、A3の音(440Hz)の他に、A4(880Hz)、E5(1320Hz)、A5(1760Hz)…というように、A3の2倍、3倍、4倍…の周波数の音が同時にでるのです。これらの音を、もとの音（基音という。この場合はA3）に対して倍音と呼びます。また、周波数が基音の2倍の音を第2倍音、3倍の音を第3倍音…と呼びます。基音、倍音はそれぞれ周波数が違う正弦波です。

周波数の違う音が同時に出れば、和音として聞こえそうなものですが、基音の整数倍の音は基音とよく混じりあうので、ひとつの音として聞こえるのです。（その場合、他の周波数にくらべてある程度のレベルであれば、もっとも低い周波数の音として聞こえるように人間の耳ができているようです。）

つまり、ひとつの音として認識されるものは、実は多くの倍音の混じりあったものなのですが、その混じり具合、つまり、第何倍音がどのくらいのレベルで含まれているかということが楽器によってそれぞれ異なっていて、それによって楽器特有の音色が作られるのです。

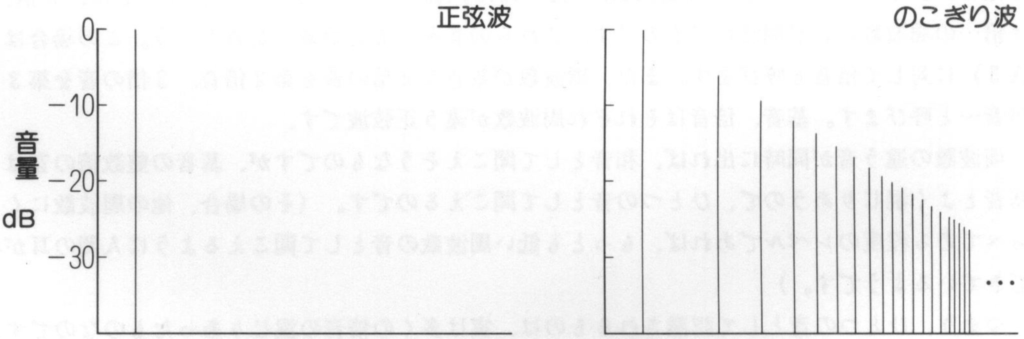
基音に対して倍音の量が多ければ明るい音になり、少なければ正弦波に近い地味な音になり

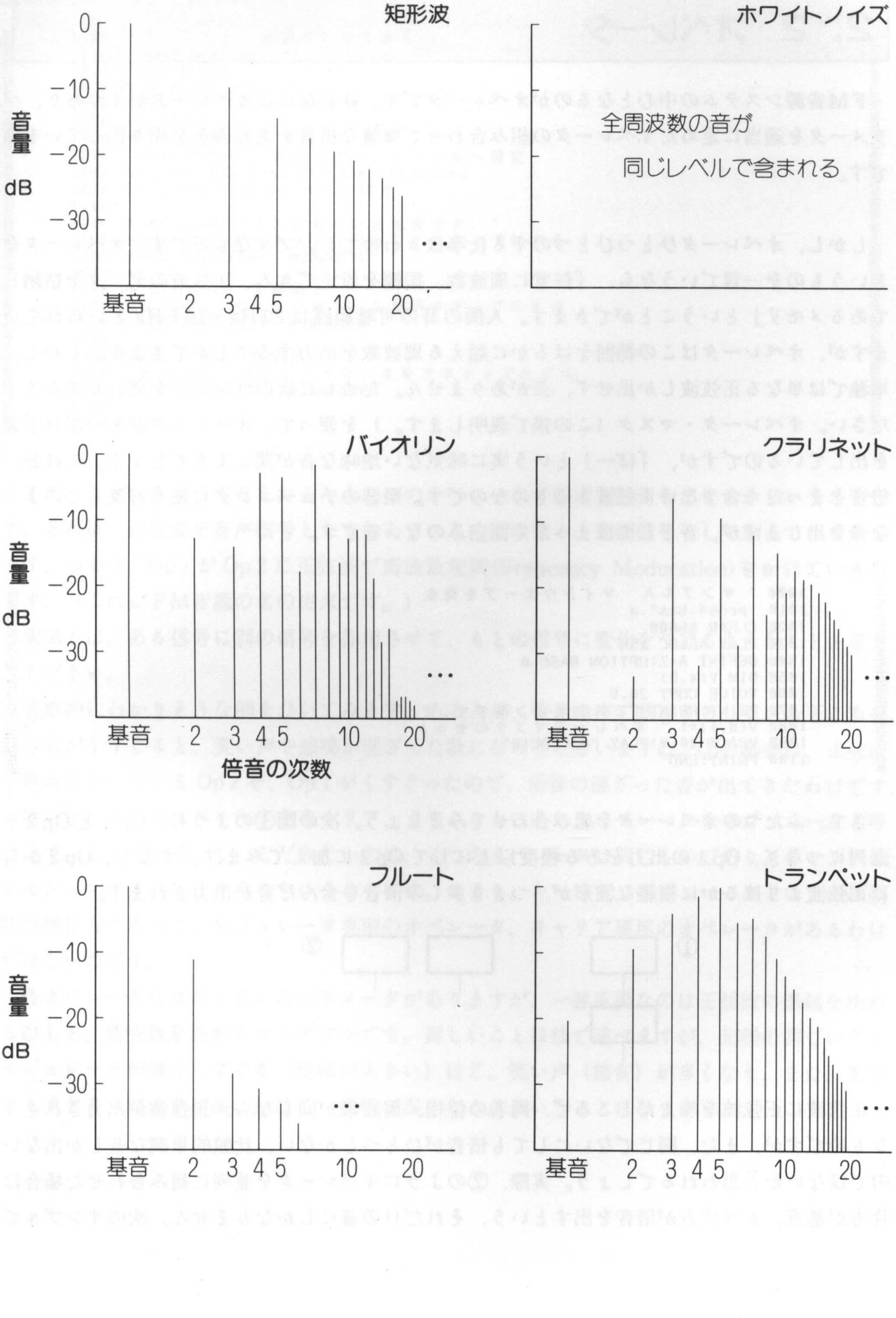
ます。倍音が多すぎると、音程感がなくなりノイズになります。また、高い周波数の倍音が多ければ高域が強調されてキラキラした音になり、第2、第3倍音が多いとおちついた音になります。

楽器音に含まれる倍音は整数倍のものがほとんどですが、ティンパニなどの打楽器や、ベル、鐘など金属的な音には非整数倍の倍音が含まれています。非整数次倍音は基音とうまく混じりあわないことが多く、音程感がなくなったり複数の音がなっているように聞こえることがあります。（このような音はいままでのアナログシンセサイザでは作れませんでした。FM音源のもっとも得意とする分野です。）

結論を述べましょう。音色は、倍音の構成、つまり第何倍音がどれくらいあるかという組み合わせによって決定されます。

倍音の構成を視覚的にとらえる方法として、音のスペクトルというものがあります。これは、縦軸にレベル（音量）、横軸に周波数をとって倍音構成をグラフ化したもので、つかみどころのない音色というものを具体的に表現する手段として有効です。例として、正弦波、アナログシンセの音源としてよく使われる矩形波、ノコギリ波、ノイズ、いくつか楽器音のスペクトルを示しておきます。





矩形波
ホワイトノイズ
全周波数の音が
同じレベルで含まれる
バイオリン
クラリネット
フルート
トランペット
音量
dB
0
−10
−20
−30
基音
2
3
4
5
10
20
…
倍音の次数

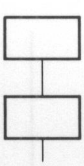
2．2　オペレータ

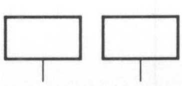
FM音源システムの中心となるのがオペレータです。ＯＰＮにはオペレータが4つあり、パラメータを適当に定めたオペレータの組み合わせで複雑な倍音を含む音色を作り出しているのです。

しかし、オペレータひとつひとつのする仕事はきわめてシンプルなものです。オペレータをというものを一言でいうなら、「任意に周波数、振幅を設定できる、正弦波のデータを格納してあるメモリ」ということができます。人間の耳の可聴範囲は20 Hz～20 KHzといわれていますが、オペレータはこの範囲をはるかに超える周波数を出力することができます。しかし、単独では単なる正弦波しか出せず、芸がありません。ためしに次のサンプルを実行してみてください。オペレータ・マスク（この後で説明します。）を使って、オペレータひとつだけで音を出しているのですが、「ぽー」という実に味気ない地味な音が聞こえるでしょう。これが、倍音をまったく含まない正弦波そのものなのです。楽器のチューニングに使う音叉もこのような音を出しますが、音としてはまったく面白みのない音です。

```
1000 ' サンプルA　サインウエーブを発生
1010 'prog4.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 20,V
1070 PRINT:PRINT "これがサイン波です。"
1080 V(0,1)=1 ' オペレータマスクのセット
1090 VOICE 20,V:PLAY "@20O6C1R"
1100 PRINT:END
```

さて、ふたつのオペレータを組み合わせてみましょう。次の図①のように、Op 1とOp 2を直列につなぎ、Op 1の出力をある程度以上にしてOp 2に加えてみます。すると、Op 2からは正弦波よりはるかに複雑な波形が、つまり多くの倍音を含んだ音が出力されます。

①



②

正弦波に正弦波を加えたところで、両者の位相、周波数が同じなら、正弦波が出力されそうなものですが、また、同じでないにしても倍音がひとつしかない、比較的単調な音しか出ないのではないかと思われるでしょう。実際、②のようにオペレータを並列に組み合わせた場合は、片方が基音、もう片方が倍音を出すという、それだけの音にしかなりません。次のサンプルで、

①と②の両方の音を聞き較べてみてください。

```
1000 ' サンプルB　変調をかけてみる
1010 'prog5.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 20,V
1070 V(0,0)=0*8+6 ' アルゴリズムをＡ１６へ設定
1080 V(1,7)=1:V(2,7)=1:V(3,7)=1:V(4,7)=2
1090 V(1,5)=20 ' 出力レベルの設定
1100 '
1110 PRINT:PRINT "これがもとの正弦波です。"
1120 V(0,1)=4 ' オペレータ３のみをアクティブにする
1130 VOICE 20,V:PLAY "@20O6C1R"
1140 '
1150 PRINT:PRINT "変調例１"
1160 V(0,1)=3 ' オペレータ１，２をアクティブにする
1170 VOICE 20,V:PLAY "@20O6C1R"
1180 '
1190 PRINT:PRINT "変調例２"
1200 V(0,1)=12 ' オペレータ３，４をアクティブにする
1210 VOICE 20,V:PLAY "@20O6C1R"
1220 '
1230 END
```

さて、なぜ、直列にするとたったふたつのオペレータで豊富な倍音を出せるかということです。それは、正弦波を音声信号として加えているのではなく、変調信号として加えているからです。つまり、Op 1 が Op 2 に正弦波で周波数変調(Frequency Moduration)をかけているのです。（これがＦＭ音源の名の由来です。）

変調とは、ある信号に別の信号を作用させて、もとの信号に変化をつけるようなことと考えてください。

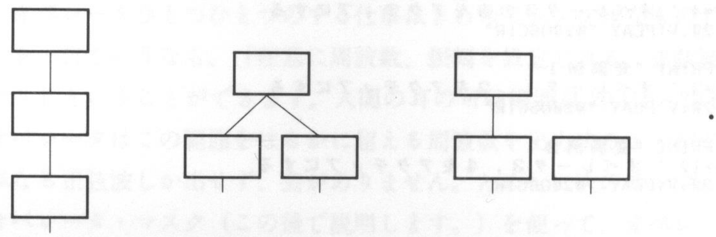
直感的にわかりそうな例をひいてみます。ある人が、普通の声で歌おうとしているところを、別の人がくすぐると、笑い声や悲鳴が混ざった歌になってしまいますね。①の場合は、正弦波で歌おうとしている Op 2 を、Op 1 がくすぐったので、倍音の混ざった音が出てきたわけです。

歌い手にあたるオペレータをキャリア、くすぐる人にあたるオペレータをモジュレータと呼びます。ここで注意していただきたいのは、４つのオペレータは同じものであり、キャリアにもモジュレータにもなりうるということです。つまり、キャリア、モジュレータというのは役割の呼び方であって、モジュレータ専用のオペレータ、キャリア専用のオペレータがあるわけではないのです。

各オペレータにはたくさんのパラメータがありますが、一番重要なのは正弦波の振幅を決めるＯＬと、周波数を決めるマルチプルです。詳しいことは後で述べますが、先程の例でいうと、モジュレータが強くくすぐる（振幅が大きい）ほど、笑い声（倍音）が多くなり、また、素早くくすぐる（周波数が高い）ほど、悲鳴が高くなる、ということになります。

2．3　アルゴリズム

これまで述べたように、最低ふたつのオペレータがあればそれを縦に並べ、上をモジュレータ、下をキャリアとしてFM音源が成立します。オペレータが2個の場合はそれだけですが、3個の場合は、縦に3段重ねたり、ひとつのモジュレータでふたつのキャリアに変調をかけたり、といったいろいろな組み合わせが考えられます。



このようなオペレータの組み合わせ方、どのオペレータをキャリアにしどれをモジュレータにするか、という組み合わせをアルゴリズムと呼んでいます。

アルゴリズムとは、もともと数学の分野の言葉で、問題解決のための手順・方法を意味します。これを、音色を作り出す手順という意味で、FM音源ではオペレータの接続を表わすのに使っています。

さて、PCシリーズの場合、オペレータが4個あり実に数多くのアルゴリズムが考えられますが、操作性なども考えて、そのうちの8種類がプリセットされています。ユーザはアルゴゴリズムナンバーを0～7の範囲で選べばよいのです。残念ながら、プリセットされていないアルゴリズムを考えても実行する方法はありませんが、実用上は差し支えないでしょう。8種のアルゴリズムについては第1章を参照してください。

一般に、複数のモジュレータがひとつのキャリアに変調をかける場合、並列に変調をかけるより直列に変調をかけるほうが、変調の効果は大きくなります。つまり、ひとつのキャリアにみっつのモジュレータがのっているAl0～Al3のうち、もっとも大きく変調がかかり倍音の多い明るい音を得られるのはAl0ということです。

アルゴリズムと音色の関係について簡単にいうと、モジュレータの数が多いほど、明るく鋭い、ソロ楽器向きの音になります。逆に、キャリアの数が多いほど、落ちついた厚みのある、アンサンブル向きの音になります。詳しくは第4章で触れることにしますので、そちらを参照してください。

2．4　オペレータ・マスク

4つあるオペレータのOn、Offを決めるものです。ただし、オペレータ・マスクはＯＰＮの機能ではありません。ＢＡＳＩＣインタプリタがマスクされているオペレータにはKey-on信号を送らないようにしているのです。

4つのオペレータについてそれぞれOn/Offの2とおりがあり、全部で16とおりの場合がありますから、指定できる値の範囲も0～15となっています。指定値と各オペレータの状態との関係は表のとおりで、10進数と2進数の対応と同じです。

指定値	Op 4	Op 3	Op 2	Op 1
0	Off	Off	Off	Off
1	Off	Off	Off	On
2	Off	Off	On	Off
3	Off	Off	On	On
4	Off	On	Off	Off
5	Off	On	Off	On
6	Off	On	On	Off
7	Off	On	On	On
8	On	Off	Off	Off
9	On	Off	Off	On
10	On	Off	On	Off
11	On	Off	On	On
12	On	On	Off	Off
13	On	On	Off	On
14	On	On	On	Off
15	On	On	On	On

通常の演奏時には、全オペレータを駆使しますから、15を設定します。他の値は、何に使用するのか疑問に思われるでしょうが、実は音作りの際にたいへん重宝するものです。

たとえば、ピアノの音はAl 4で、Op 1+Op 2がアタックのコツンという音を、Op 3+Op 4が基本的な音を表わすようにして作ることが多いのですが、このような音を作る際には、まずマスクを3にしてOp 1+Op 2だけについてエディトし、次にマスクを12にしてOp 3+Op 4についてエディトを行なうようにします。コツンという音を作っていく途中では、Op 3+Op 4は黙っているほうが作業がしやすいのです。

```
1000 ' サンプルC  オペレータをマスクする
1010 'prog6.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 13,V
1070 '
1080 PRINT:PRINT "これがもとのピアノサウンドです。"
1090 VOICE 13,V:PLAY "@13O6C1R"
1100 '
1110 PRINT:PRINT "オペレータ3とオペレータ4をマスクしました。"
1120 V(0,1)=3
1130 VOICE 13,V:PLAY "@13O6C1R"
1140 '
1150 PRINT:PRINT "オペレータ1とオペレータ2をマスクしました。"
1160 V(0,1)=12
1170 VOICE 13,V:PLAY "@13O6C1R"
1180 '
1190 END
```

また、こうすると、アルゴリズムの各部でどのような音ができ、それが集まって最終的にどのような音ができるか、という流れがはっきり把握できます。従来のアナログシンセサイザがフィルタを用いて倍音をカットする、いわば減算方式で音を作っていたのに対して、FM音源シンセサイザでは加算方式で音を作る点が特徴となっていますので、オペレータマスクの持つこのような教育的効果は重要です。

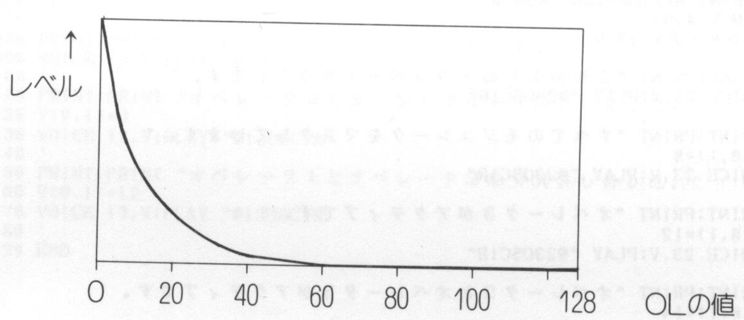
もうひとつ、プリセットボイスを解析する場合にも、次のように使うと有効です。

まず、キャリアのみをOnにして音を出してみます。（具体的には、Al 0～Al 3に対しては Mask = 8、Al 4に対してはMask = 10、Al 5～Al 6に対してはMask = 14、Al 7に対しては Mask = 15とします。）　モジュレータがないので、変調のかからない、正弦波が聞こえるはずです。そこで、モジュレータをひとつづつOnにしながら、音を聞き較べてみましょう。プリセットボイスのデータと合わせて見れば、音色が生み出される仕組みがさらにはっきり理解できると思います。次のサンプルは、音色番号23－TRUMPETを使用したものです。

```
1000 ' サンプルD　オペレータをマスクする
1010 'prog7.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 23,V
1070 '
1080 PRINT:PRINT "これがもとのトランペットサウンドです。"
1090 VOICE 23,V:PLAY "@23O5C1R"
1100 '
1110 PRINT:PRINT "すべてのモジュレータをマスクしています。"
1120 V(0,1)=8
1130 VOICE 23,V:PLAY "@23O5C1R"
1140 '
1150 PRINT:PRINT "オペレータ3がアクティブです。"
1160 V(0,1)=12
1170 VOICE 23,V:PLAY "@23O5C1R"
1180 '
1190 PRINT:PRINT "オペレータ2とオペレータ3がアクティブです。"
1200 V(0,1)=14
1210 VOICE 23,V:PLAY "@23O5C1R"
1220 '
1230 PRINT:PRINT "すべてのモジュレータをアクティブにしています。"
1240 V(0,1)=15
1250 VOICE 23,V:PLAY "@23O5C1R"
1260 '
1270 END
```

2．5 アウトプットレベル

アウトプットレベル（ＯＬ）は、各オペレータの出力の振幅を決めるパラメータで、第１章ではトータルレベルと呼んでいたものです。ＰＣでは０～127の範囲で指定できます。注意すべき点は、０のとき振幅が最大になるということです。つまり、アウトプットレベルは減衰量を指定するものと思ってください。また、数値と実際の出力レベルとは対数的な関係にあり、数値が８～９増えるごとにレベルは1/2になります。最大レベルの半分ぐらいの音を出したければ、ＯＬは８か９に設定すればいいわけです。127の半分のつもりでＯＬ＝63などとすると、最大レベルの約1/100となり、音が出ないといってあわてることになってしまいます。




```
1000 ' サンプルE  出力レベルを変化させる
1010 'prog8.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 21,V
1070 '
1080 PRINT:PRINT "これがもとのブラスサウンドです。"
1090 VOICE 21,V:PLAY "@21O5C1R"
1100 '
1110 FOR OL=0 TO 127 STEP 8
1120   PRINT "出力レベル=";OL
1130   VOICE REG &H4C,OL ' オペレータ4の出力レベルをセット
1140   GOSUB *PLAY.
1150 NEXT
1160 END
1170 '
1180 *PLAY.
1190 VOICE REG &H28,&HF0
1200 FOR TIMING=1 TO 700:NEXT
1210 VOICE REG &H28,0
1220 FOR TIMING=1 TO 1000:NEXT
1230 RETURN
```

アウトプットレベルはキャリアに対しては音量を表わし、値が小さいほど大きな音になります。モジュレータに対してはキャリアにかける変調の度合いを表わします。値が小さいほど、大きく変調がかかりますので、倍音が増え、明るい音になります。あまり変調をかけすぎるとノイズになってしまいますので、適当に加減してください。特にフィードバック付きのオペレータは過変調になりやすいので注意が必要です。

```
1000 ' サンプルF  出力レベルを変化させる
1010 'prog9.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 29,V:NV=V(3,5)
1070 '
1080 PRINT:PRINT "これがもとのギターサウンドです。"
1090 VOICE 29,V:PLAY "@29O5C1R"
1100 '
1110 FOR OL=0 TO 64 STEP 8
1120   PRINT "出力レベル=";OL
1130   V(3,5)=OL ' オペレータ3の出力レベルをセット
1140   VOICE 29,V:PLAY "@29O5C1R"
1150 NEXT
1160 '
1170 PRINT:PRINT "再びもとのギターサウンドです。"
1180 V(3,5)=NV
1190 VOICE 29,V:PLAY "@29O5C1R"
1200 '
1210 END
```

モジュレータのアウトプットレベルが倍音の量を変化させるといっても、たとえば第4倍音を大きくする、というように個々の倍音について指定できるわけではありません。どの倍音を発生させるかということはマルチプルで指定できますので、それと組み合わせて使うことによって、ほとんど無限に多くの音色を生み出すことができます。

2．6 フィードバック

フィードバックは、効果音に用いるホワイトノイズなどを作るとき、ストリングス系の高域のサラサラした音を作るとき、ブラス系のアタック時の鋭い変調を作るときなどに不可欠なもので、FM音源ならではの機能といえるでしょう。

フィードバックとは、オペレータの出力を自分自身にもどして、自分で自分に変調をかけることをいい、どのアルゴリズムにもひとつずつフィードバックのかかるオペレータがあります。オペレータの上にモジュレータがのっていると考えればよく、要するにひとつのオペレータをふたつぶんに使っているわけです。

ですから、フィードバックのかかったオペレータをキャリアとして使うと、それだけでも正弦波以外の音を発生することができます。が、そのような使い方をすることは少なく、普通はより強力な変調のかかるモジュレータとして使用します。実際、フィードバックのかかったオペレータをキャリアとしているアルゴリズムはAl7だけです。

次のサンプルは、音色番号20－ＳＩＮ　ＷＡＶＥ（Al7）で他のキャリアをすべてマスクし、フィードバック付のオペレータのみを発音させています。

```
1000 'サンプルG　フィードバック
1010 'prog10.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 20,V:ALG=V(0,0) AND &H7
1070 '
1080 PRINT:PRINT "これがふつうの正弦波です。"
1090 VOICE 20,V:PLAY "@20O6C1R"
1100 '
1110 PRINT:PRINT "オペレータ1のみがアクティブです。"
1120 V(0,1)=1
1130 FOR FB=0 TO 7
1140   PRINT "フィードバック量=";FB
1150   V(0,0)=FB*8+ALG ' フィードバックをセット
1160   VOICE 20,V:PLAY "@20O6C1R"
1170 NEXT
1180 '
1190 END
```

さて、普通のモジュレータとキャリアとの関係では、変調の度合いを決めるのはモジュレータのＯＬでした。フィードバックのかかるオペレータの変調度を決めるのが、Feedback（Ｆｂと略する）です。ＰＣの場合、Ｆｂは０～７の範囲で、０のときは変調がまったくなく、７のとき最大になります。

フィードバックの値と音色の関係は、モジュレータのアウトプットレベルと音色との関係とほぼ同じです。つまり、フィードバックを大きくすると、高調波成分が多くなり、明るい感じの音になりますが、大きすぎるとかえって下品な感じになってしまいます。ただし、そのオペレータのアウトプットレベルによってフィードバックの影響力は変わりますから、その点にも注意してください。

```
1000 ' サンプルＨ　フィードバック
1010 'prog11.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 VOICE INIT
1060 DIM V(4,9)
1070 VOICE COPY 53,V:NV=V(0,0):ALG=V(0,0) AND &H7
1080 '
1090 PRINT:PRINT "これがふつうの蒸気機関車サウンドです。"
1100 VOICE 53,V:PLAY "@53O6C1R"
1110 '
1120 FOR FB=7 TO 0 STEP -1
1130   PRINT "フィードバック量=";FB
1140   V(0,0)=FB*8+ALG ' フィードバックをセット
1150   VOICE 53,V:PLAY "@53O6C1R"
1160 NEXT
1170 '
1180 PRINT:PRINT "再びもとの蒸気機関車サウンドです。"
1190 V(0,0)=NV
1200 VOICE 53,V:PLAY "@53O6C1R"
1210 '
1220 END
```

2. 7 マルチプル

各オペレータの出力する正弦波の周波数を決めるパラメータです。0～15の範囲で指定しますが、0は便宜上の値で、実際には0.5として処理されます。

少し詳しく説明すると、BLOCK/F-Numberによって指定された音程の何倍の周波数を出力するかを決めるものです。たとえば、BLOCK/F-NumberがA 3 (440 Hz)を指定しているとします。このとき、マルチプルが1であれば440 Hz、0であれば220 Hz（つまり、オクターブ下のA 2です）、8であれば3520 Hz（3オクターブ上のA 6）が実際に発音されます。

```
1000 ' サンプルI  マルチプル
1010 'prog12.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 20,V
1070 '
1080 PRINT:PRINT "これがふつうの正弦波です。"
1090 VOICE 20,V:PLAY "@20O5A1R"
1100 '
1110 FOR ML=0 TO 15
1120   PRINT "マルチプル=";ML
1130   V(1,7)=ML ' オペレータ1のマルチプルをセット
1140   VOICE 20,V:PLAY "@20O5A1R"
1150 NEXT
1160 '
1170 END
```

なぜこのような面倒なことをするのか不思議に思われる方が多いと思います。ここで2．1節をもういちど見て、音色とは何か思い出してください。音色とは倍音の組み合わせです。つまり、ある特定の音色を作るためには、倍音を自由に作り出せることが必要なのです。

さて、マルチプルの値によって実際にどのような倍音が出るか、そして音色はどうなるのか、具体例を挙げてみましょう。

まず、キャリアのマルチプルをc、モジュレータのマルチプルをmとします。また、$c=m$または$2c \leqq m$とします。こうするとキャリアのマルチプルがそのまま基音の周波数を表わすようになりますし、また実際問題としてそうでない設定はほとんど行なわれません。

そこで、出力される倍音の次数を求める方法を考えてみましょう。キャリアの出力する（つまり基音の）周波数をfc、モジュレータの出力する周波数をfmとすると、最終的に出力される周波数fは、

$$f=|fc \pm n \cdot fm| = T \cdot |c \pm n \cdot m|$$

となります。ただし、TはBLOCK/F-Numberで指定された周波数、nは0以上の整数です。（fの答えはひとつではありません。）

倍音の次数とは、f/fcのことですから、次数xは次の式で表わされます。

$$x = \left| 1 + \frac{n \cdot m}{c} \right| \qquad (nは整数)$$

それでは具体的にcとmの値と発生する倍音の次数を調べてみましょう。

○c：m＝1：1　のとき

$n = 0 \rightarrow x = 1$　(基音)
$n = 1 \rightarrow x = 2$　(第2倍音)
$n = 2 \rightarrow x = 3$　(第3倍音)
$n = 3 \rightarrow x = 4$　(第4倍音)

このようにすべての整数次倍音が発生します。アウトプットレベルが適切なら、のこぎり波が得られます。さまざまな音色に使われ、特にブラスやストリングスの音に適しています。

○c：m＝1：2　のとき

$n = 0 \rightarrow x = 1$　(基音)
$n = 1 \rightarrow x = 3$　(第3倍音)
$n = 2 \rightarrow x = 5$　(第5倍音)
$n = 3 \rightarrow x = 7$　(第7倍音)

このようにすべての奇数次倍音が含まれます。アウトプットレベルが適切なら、矩形波が得られます。木管楽器などおちついた音によく使われる設定です。

○c：m＝1：3　のとき

$n = 0 \rightarrow x = 1$　(基音)
$n = 1 \rightarrow x = 4$　(第4倍音)　　$n = -1 \rightarrow x = 2$　(第2倍音)
$n = 2 \rightarrow x = 7$　(第7倍音)　　$n = -2 \rightarrow x = 4$　(第4倍音)
$n = 3 \rightarrow x = 10$　(第10倍音)　　$n = -3 \rightarrow x = 8$　(第8倍音)

このように3の倍数次の倍音がかけた音になります。ハープ・ギターなど撥弦楽器によく使われる設定です。

```
1000 ' サンプルJ  マルチプル
1010 'prog13.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0
1050 DIM V(4,9)
1060 VOICE COPY 20,V
1070 '
1080 V(0,0)=0*8+4 ' アルゴリズム４にセット
1090 V(0,1)=12    ' オペレータ３とオペレータ４がアクティブ
1100 V(3,5)=90    ' モジュレータの出力レベル
1110 '
1120 PRINT:PRINT "キャリア：モジュレータ = 1:1"
1130 V(3,7)=1:V(4,7)=1
1140 VOICE 20,V:PLAY "@20O6C2R"
1150 '
1160 PRINT:PRINT "キャリア：モジュレータ = 1:2"
1170 V(3,7)=2:V(4,7)=1
1180 VOICE 20,V:PLAY "@20O6C2R"
1190 '
1200 PRINT:PRINT "キャリア：モジュレータ = 1:3"
1210 V(3,7)=3:V(4,7)=1
1220 VOICE 20,V:PLAY "@20O6C2R"
1230 '
1240 END
```

2．8　エンベロープジェネレータ

音を決定するみっつの要素として、音程、音量、音色を挙げましたが、これだけではまだ不充分です。というのは、これらの要素が時間的に一定であるとはかぎらないからです。楽器音の場合、音程については一定でないと困りますが、ほとんどの楽器音は音量、音色が時間的に変化します。

もっともわかりやすいところで、ピアノの音を例にとって説明しましょう。ピアノを弾くと、鍵盤を叩いた瞬間に音量がもっとも大きくなり、そのあと鍵盤を押している間はゆっくりと小さくなっていきます。これがピアノの音量の時間的変化です。

では、音色の時間的変化というと、耳で聞いただけではわかりにくいのですが、波形をオシロスコープで見てみると、打鍵直後は倍音の多い複雑な波形をしていて、次第に倍音が減り正弦波に近い波形になっていくことがわかります。つまり、明るい音がだんだんと丸い音に変わっていく、という変化をするのです。

このような時間的変化をエンベロープといい、エンベロープをコントロールする、あるいは作り出すものをエンベロープジェネレータ（EG）といいます。
エンベロープが音の印象に与える影響は意外に大きく、音色がさほど似ていなくてもエンベロープが似ていると、同じような音に聞こえてしまうほどです。次のサンプルは、音色番号28－EPIANO3と音色番号33－EORGAN2を例にとり、互いにエンベロープを入れ替えてみたものです。

```
1000 ' サンプルK  エンベロープ
1010 'prog14.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0:VOICE INIT
1050 DIM V(4,9),VH(4,9),VX(4,9)
1060 VOICE COPY 28,VH ' 電子ピアノ3
1070 VOICE COPY 33,VX ' 電子オルガン2
1080 '
1090 PRINT:PRINT "これがノーマルの電子ピアノです。"
1100 VOICE 28,VH:PLAY "@28C2R2"
1110 '
1120 PRINT:PRINT "これが電子オルガンのエンベロープを持った電子ピアノです。"
1130 VOICE COPY 28,V
1140 FOR OP=1 TO 4
1150   FOR I=0 TO 4
1160     V(OP,I)=VX(OP,I)
1170   NEXT
1180 NEXT
1190 '
1200 VOICE 28,V:PLAY "@28C2R2"
1210 '
1220 PRINT:PRINT "これがノーマルの電子オルガンです。"
1230 VOICE 33,VX:PLAY "@33C2R2"
1240 '
1250 PRINT:PRINT "これが電子ピアノのエンベロープを持った電子オルガンです。"
1260 VOICE COPY 33,V
1270 FOR OP=1 TO 4
1280   FOR I=0 TO 4
1290     V(OP,I)=VH(OP,I)
1300   NEXT
1310 NEXT
1320 '
1330 VOICE 33,V:PLAY "@33C2R2"
1340 '
1350 PRINT:PRINT "これがノーマルの電子ピアノです。"
1360 VOICE 28,VH:PLAY "@28C2R2"
1370 '
1380 END
```

それでは、エンベロープジェネレータのパラメータについて説明しましょう。

○AR

key-onから、最大レベルに達するまでの速度。つまり、音の立ち上がりを決めるパラメータ。0～31の範囲で指定し、数値が大きくなるほど速くなる。（0の場合、音が出ない）

○DR

最大レベルから、サスティンレベルにまで下がる速度。0～31の範囲で指定し、数値が大きくなるほど速くなる。（0の場合、SR、SLにかかわらず最大レベルのまま持続する）

○SR

サスティンレベルから、音が消えるまで下がる速度。0～31の範囲で指定し、数値が大きくなるほど速くなる。0の場合はサスティンレベルのまま持続する。

○RR

key-off後、音が消えるまでの速度。つまり、余韻が残る時間を決めるパラメータ。0～15の範囲で指定し、数値が大きくなるほど速くなる。（0の場合でも、やはり音は消えていく）

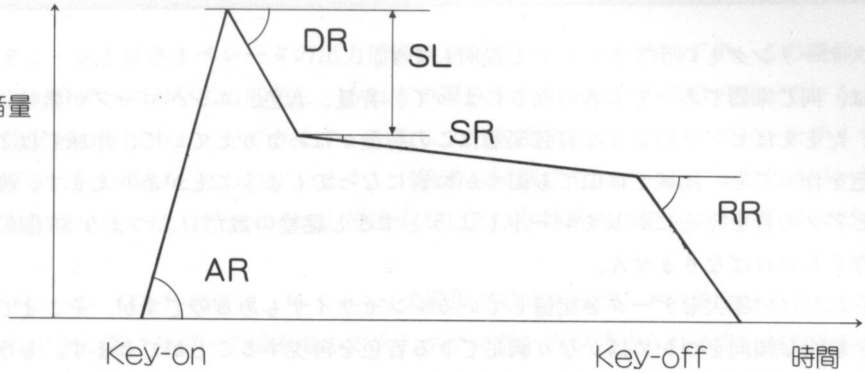
○SL

サスティンレベル。アタック後のディケイの部分での減衰量を表わす。0～15の範囲で指定し、数値が大きいほどレベルは低くなる。（数値が2大きくなるごとにレベルが半分になる）

Rのつく各パラメータはRate、すなわち速度ですから、数値が大きいほど時間的には短くなります。普通、速度は距離を時間で割ったものですが、この場合距離に相当するものはレベルの差です。つまり、Rateの値が同じでも、ＳＬの値が異なればレベルの差が異なりますから、変化にかかる時間も変わってきます。ですから、ＤＲ、ＳＲはＳＬとのバランスを考えて設定しなければなりません。

各パラメータとエンベロープの関係を図に示します。エンベロープは通常縦軸にレベル、横

軸に時間をとったグラフとして表わされますから、Rateが角度に対応することに注意してください。



エンベロープはよっつのオペレータについて、それぞれ独立に設定できます。各オペレータのアウトプットレベルが独自に変化するわけですが、キャリアに設定したエンベロープは音量の時間的変化になり、モジュレータに設定したエンベロープは、倍音の量、つまり音色の時間的変化を作り出すのです。

ところで、エンベロープはアウトプットレベルの時間的変化を作り出すものですから、ＯＬに対する相対値によって定まっています。つまり、エンベロープの最大値やＳＬは、ＯＬに対する相対値を表わしているのです。

ですから、同じエンベロープを設定しても、ＯＬが違えば実際に出力されるレベルが変わってきます。キャリアの場合はさほど問題にならないのですが、モジュレータの場合、音色に直接影響しますので、注意が必要です。たとえば、微妙な音色変化を出したいときにはモジュレータのＯＬをやや小さめの値にしてください。

つまり、モジュレータのＯＬは、エンベロープをどれだけ効かせるかを決めるパラメータともいえるわけです。ＯＬとＥＧは常にバランスをとって決めなければいけません。

2.9 スケーリング

まず、スケーリングということについて説明します。

楽器音は、同じ楽器であっても音の高さによって、音量、音色、エンベロープが微妙に異なるのです。たとえばピアノのような打弦楽器はこの特徴がはっきりしていて、中域ではよく似ている音色を作っても、高域では似ても似つかぬ音になってしまうことがありえます。極端に言えば、ピアノの音を完全にシュミレートしようとすると鍵盤の数だけ、つまり88個の音色データを作らなければなりません。

実際にそれだけの膨大なデータを記憶しているシンセサイザもあるのですが、そこまでしなくとも、全般的な傾向をつかめばかなり満足できる音色を再現することができます。もういちどピアノを例にとると、高い音ほどエンベロープの変化が速くなる傾向があります。（これは、ピアノが打弦楽器であり、高音ほど弦が短くなることに由来します。）そこで、高い音ほどエンベロープ変化が速くなるように補正してやるわけです。

このように音域によって異なる楽器音を原音に忠実に、しかもなるべく簡単に再現できるように、補正を加える機能をスケーリングといいます。スケーリングには前述のようにエンベロープの長さを補正するRate Scalingの他、音量や音色を補正するLevel Scalingがあります。ただし、ＰＣシリーズにはLevel Scalingの機能はありません。

レイトスケールは0～3の値で指定し、数値が大きいほど、補正量が大きくなり、高音でエンベロープが速くなる度合いが増します。0のときは、音程にかかわらず一定の速さのエンベロープとなります。

```
1000 ' サンプルL  キースケーリング
1010 'prog15.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z:OPTION BASE 0:VOICE INIT
1050 DIM V(4,9)
1060 VOICE COPY 13,V
1070 '
1080 PRINT:PRINT "これがノーマルのピアノです。"
1090 PLAY "@13O3CRO7CR2"
1100 '
1110 PRINT:PRINT "キースケーリングを行わないようにします。"
1120 FOR OP=1 TO 4
1130   V(OP,6)=0
1140 NEXT
1150 '
1160 VOICE 13,V:PLAY "O3CRO7CR2"
1170 '
1180 END
```

2. 10 デチューン

デチューンとは各オペレータの出力周波数をわずかにずらす機能です。この機能により、エレピ、ストリングスなどの音色に、うねりや厚みを与えて複数の楽器が鳴っているようなコーラス効果を生み出すことができます。一般にFM音源の音は、アナログシンセサイザの音に較べて厚みに欠ける、と言われています。（筆者も同感です。） この欠点をカバーし、リアルな音を作るためには、デチューンと後述するＬＦＯを効果的に使うことが不可欠です。

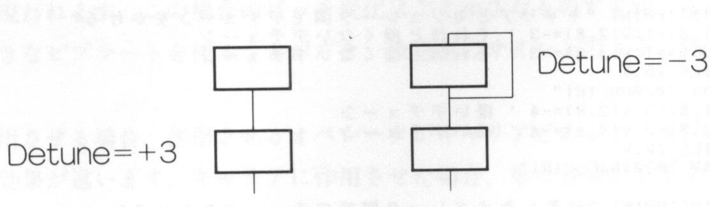
デチューン（Dt）の値は、－４～＋３の範囲で設定します。＋の値を与えるともとの周波数よりわずかに高い音が－の値を与えるとわずかに低い音が発せられます。絶対値が大きいほどずれも大きくなりますが、最大でも半音の数十分の１程度の小さなものです。

デチューンのかけ方にも、キャリアどうしにかけたり、キャリアとモジュレータの間にかけたり、目的に応じて、いろいろな定石があります。

もっとも簡単なのは複数のキャリア間でデチューンの値を変えて、ピッチをずらす方法です。この方法を用いるとまさしく複数の音源が鳴っている状態になり、周期的音量変化も得られることになります。しかし、むやみにデチューンを大きくすると、単にチューニングがはずれたようなことになり、音にうねりがでてくることがありますので、注意してください。

次に、キャリアとそれに対するモジュレータの間にデチューンをかける方法があります。この方法を用いると、キャリアに対してモジュレーションの位相がゆれ動くことになりますから、周期的な音色変化が起こります。楽器のエフェクターでいうと、フェイザーのような、コーラスよりもしっとりした効果が得られます。ただし、モジュレータのアウトプットレベルが大きすぎると耳あたりの悪い音になりますので注意してください。

もっとも効果的な方法は、キャリアと、別のキャリアに対するモジュレータの間にデチューンをかける方法です。具体的には、アルゴリズムの図上で斜めの位置にあるオペレータどうしのデチューンの値を、絶対値を等しく符号を逆にします。



こうすると、キャリアの間にデチューンをかけた場合に較べて、不必要なうねりが消えて、分厚いきれいなコーラスが得られます。ただ、この方法が使えるアルゴリズムは4と6だけですので、その他のアルゴリズムには前述の2つの方法を使うしかありません。

次のサンプルは、Al 4を使い、マルチプルの項で説明した方法によって2系統の矩形波を発生させ、それにデチューンをかけるものです。

```
1000 ' サンプルM  デチューン
1010 'prog16.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080
1090   音色データのセット
1100 '
1110 V(0,0)=0*8+4 ' アルゴリズムをAL4にセット
1120 V(0,1)=15     ' すべてのオペレータを使用
1130 V(0,2)=0      ' 矩形波を発生
1140 V(0,7)=0      ' LFOをマスク
1150 '
1160 ' オルガンのようなエンベロープにセット
1170 '
1180 FOR OP=1 TO 4
1190   V(OP,0)=10' ARを最大に
1200   V(OP,1)=0  ' DRなし
1210   V(OP,2)=31 ' SRを最大に
1220   V(OP,3)=15 ' RRを最大に
1230   V(OP,4)=15 ' SLを最大に
1240   V(OP,9)=0  ' LFOをマスク
1250 NEXT
1260 '
1270 V(1,5)=30:V(3,5)=30 ' モジュレータの出力レベルをセット
1280 V(2,5)=127 :V(4,5)=127 ' キャリアの出力レベルをセット
1290 V(1,7)=1 :V(3,7)=1 ' モジュレータのマルチプルをセット
1300 V(2,7)=1 :V(4,7)=1 ' キャリアのマルチプルをセット
1310 '
1320 PRINT:PRINT "これがもととなる音色です。"
1330 VOICE 20,V
1340 PLAY "@20O6C1R1"
1350 '
1360 PRINT:PRINT "キャリア間でデチューンをかける"
1370 V(2,8)=1:V(4,8)=-3 ' それほど深くないデチューン
1380 V(1,8)=1:V(3,8)=-3 ' それほど深くないデチューン
1390 VOICE 20,V
1400 PLAY "@20O6C1R1"
1410 V(2,8)=3:V(4,8)=-4 ' 深いデチューン
1420 V(1,8)=3:V(3,8)=-4 ' 深いデチューン
1430 VOICE 20,V
1440 PLAY "@20O6C1R1"
1450 '
1460 PRINT:PRINT "キャリアとモジュレータ間でデチューンをかける"
1470 V(1,8)=1:V(2,8)=-3 ' それほど深くないデチューン
1480 V(3,8)=1:V(4,8)=-3 ' それほど深くないデチューン
1490 VOICE 20,V
1500 PLAY "@20O6C1R1"
1510 V(1,8)=3:V(2,8)=-4 ' 深いデチューン
1520 V(3,8)=3:V(4,8)=-4 ' 深いデチューン
1530 VOICE 20,V
1540 PLAY "@20T64O5C1R1"
1550
1560 PRINT:PRINT "交差したオペレータ間でデチューンをかける"
1570 V(1,8)=1:V(4,8)=-3 ' それほど深くないデチューン
1580 V(3,8)=1:V(2,8)=-3 ' それほど深くないデチューン
1590 VOICE 20,V
1600 PLAY "@20O6C1R1"
1610 V(1,8)=3:V(4,8)=-4 ' 深いデチューン
1620 V(3,8)=3:V(2,8)=-4 ' 深いデチューン
1630 VOICE 20,V
1640 PLAY "@20T64O5C1R1"
1650 '
1660 END
```

2. 11 LFO

2. 11. 1 LFOとはなにか

ＬＦＯとは、Low Frequency Oscillator、つまり低周波の発振器のことです。これまで述べてきた各機能は基本的な音を作るためのものでした。ＬＦＯはその基本的な音に、ビブラート、ワウワウ、トレモロ、その他さまざまな効果を与えるものです。

このような効果が音のイメージに与える影響はとても大きいものがあります。ある楽器音をシミュレートするとき、その楽器でビブラートなどの効果がごく普通の演奏法であれば、シミュレート音でもその効果を出さなければなりません。そうでなければ音色自体は似ていても、その楽器の音には聞こえないのです。次のサンプルで、フルート（音色番号36）にＬＦＯがかかっていないとどうなるか、聞いてみてください。

```
1000 ' サンプルN  ＬＦＯ
1010 'prog17.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 36,V
1090 '
1100 PRINT:PRINT "これがノーマルのフルートサウンドです。"
1110 PLAY "@3605C1R"
1120 '
1130 PRINT:PRINT "ＬＦＯなしのフルートです。"
1140 V(0,7)=0
1150 FOR OP=1 TO 4
1160   V(OP,9)=0
1170 NEXT
1180 VOICE 36,V
1190 PLAY "@36C1R"
1200 '
1210 END
```

ＬＦＯは、周期的な波を発生し、これをオペレータ出力の音程（ピッチ）または音量（アンプリチュード）に作用させることです。

たとえば音程に作用させると、出力の音程がＬＦＯの波にあわせて上下することになります。これがPM（ピッチモジュレーション）で、ビブラートを作り出します。ビブラートは弦楽器には普通に現われます。この場合のピッチ変化はごく小さなものですが、1オクターブをこえるような大きなビブラートを作ることができ、救急車のサイレンのような効果音に利用されています。

音量に作用させる場合、作用させるオペレータがキャリアであるか、モジュレータであるかによって、効果が違います。キャリアに作用させた場合、単に音量が上下するだけですから、トレモロ効果となります。モジュレータに作用させると、これはモジュレータのOlが上下することと同じですから、結果としては音色が周期的に変化するワウワウ効果となります。この2つがAM（アンプリチュードモジュレーション）の効果です。

PMは各オペレータに共通ですが、AMは上記の理由でオペレータごとに設定することができます。

簡単にまとめておきます。

①	ビブラート	音程の周期的変化	PMで実現する
②	ワウワウ	音色の周期的変化	モジュレータのAMで実現する
③	トレモロ	音量の周期的変化	キャリアのAMで実現する

注意すべきことは、ＬＦＯの効果があるのはKey-onからKey-offの間だけであるということです。つまり、Key-off後の余韻の部分にはＬＦＯがかかりませんので、うまく処理しないと不自然に聞こえてしまいます。

以上がＬＦＯの概要です。次の節から、ＬＦＯの各パラメータについて、それぞれ説明することにします。

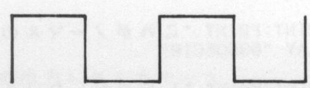
2. 11. 2 Wave Form

ＬＦＯが発生する低周波の波形を決めるパラメータで、０～５の範囲で指定します。値と波形との関係は、次のとおりです。

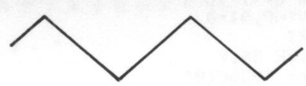
0 ： ノコギリ波



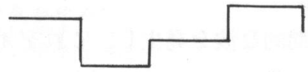
1 ： 矩形波



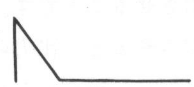
2 ： 三角波



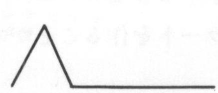
3 ： サンプル＆ホールド（ランダム波）



4 ： ノコギリ波ワンショット



5 ： 三角波ワンショット



このうち、ビブラート効果にもっとも適しているのは三角波です。サンプル＆ホールドはランダムな変化が作り出せるため、効果音によく使われます。また、ノコギリ波はSyncをOnにし、AMDを負の値にして用いるとシンセドラムなどに利用できます。２つのワンショットは、頭にピークを持つ音に効果的です。次のサンプルは、音色番号36－FLUTEのWave formをいろいろと変えてみるプログラムです。

```
1000 ' サンプルO　ＬＦＯの波型
1010 'prog18.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 DIM WAVE$(5)
1090 VOICE COPY 36,V
1100 '
1110 WAVE$(0)="のこぎり波":WAVE$(1)="矩形波":WAVE$(2)="三角波":WAVE$(3)="S & H"
1120 WAVE$(4)="のこぎり波ワンショット":WAVE$(5)="三角波ワンショット"
1130 PRINT:PRINT "これがノーマルのフルートサウンドです。"
1140 PLAY "@3604A1R"
1150 '
1160 FOR WV=0 TO 5
1170   V(0,2)=WV
1180   PRINT "ＬＦＯ波型=";WAVE$(WV)
1190   VOICE 36,V
1200   PLAY "@36A1R"
1210 NEXT
1220 '
1230 END
```

この例では違いがあまりはっきりとはわかりませんので、音色番号20－ＳＩＮ　ＷＡＶＥを使ったサンプルを次に示しておきます。

```
1000 ' サンプルＰ　ＬＦＯ波型／ＡＭ
1010 'prog19.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 DIM WAVE$(5)
1090 VOICE COPY 20,V
1100 '
1110 WAVE$(0)="のこぎり波":WAVE$(1)="矩形波":WAVE$(2)="三角波"
1120 WAVE$(3)="S & H":WAVE$(4)="のこぎり波ワンショット"
1130 WAVE$(5)="三角波ワンショット"
1140 '
1150 PRINT:PRINT "これがノーマルの正弦波です。"
1160 PLAY "@2005A1R"
1170 '
1180 FOR OP=1 TO 4
1190   V(OP,9)=4
1200 NEXT
1210 '
1220 V(0,3)=1
1230 V(0,4)=1000
1240 V(0,5)=0
1250 V(0,6)=100
1260 V(0,7)=0
1270 '
1280 FOR WV=0 TO 5
1290   V(0,2)=WV
1300   PRINT "ＬＦＯ波型=";WAVE$(WV)
1310   VOICE 20,V
1320   PLAY "@20A1R"
1330 NEXT
1340 '
1350 END
```

```
1000 ' サンプルQ　LFO波型／PM
1010 'prog20.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 DIM WAVE$(5)
1090 VOICE COPY 20,V
1100 '
1110 WAVE$(0)="のこぎり波":WAVE$(1)="矩形波":WAVE$(2)="三角波"
1120 WAVE$(3)="S & H":WAVE$(4)="のこぎり波ワンショット"
1130 WAVE$(5)="三角波ワンショット"
1140 '
1150 PRINT:PRINT "これがノーマルの正弦波です。"
1160 PLAY "@2005A1R"
1170 '
1180 FOR OP=1 TO 4
1190   V(OP,9)=1
1200 NEXT
1210 '
1220 V(0,3)=1
1230 V(0,4)=1500
1240 V(0,5)=100
1250 V(0,6)=0
1260 V(0,7)=7
1270 '
1280 FOR WV=0 TO 5
1290   V(0,2)=WV
1300   PRINT "LFO波型=";WAVE$(WV)
1310   VOICE 20,V
1320   PLAY "@20A1R"
1330 NEXT
1340 '
1350 END
```

2. 11. 3 Speed

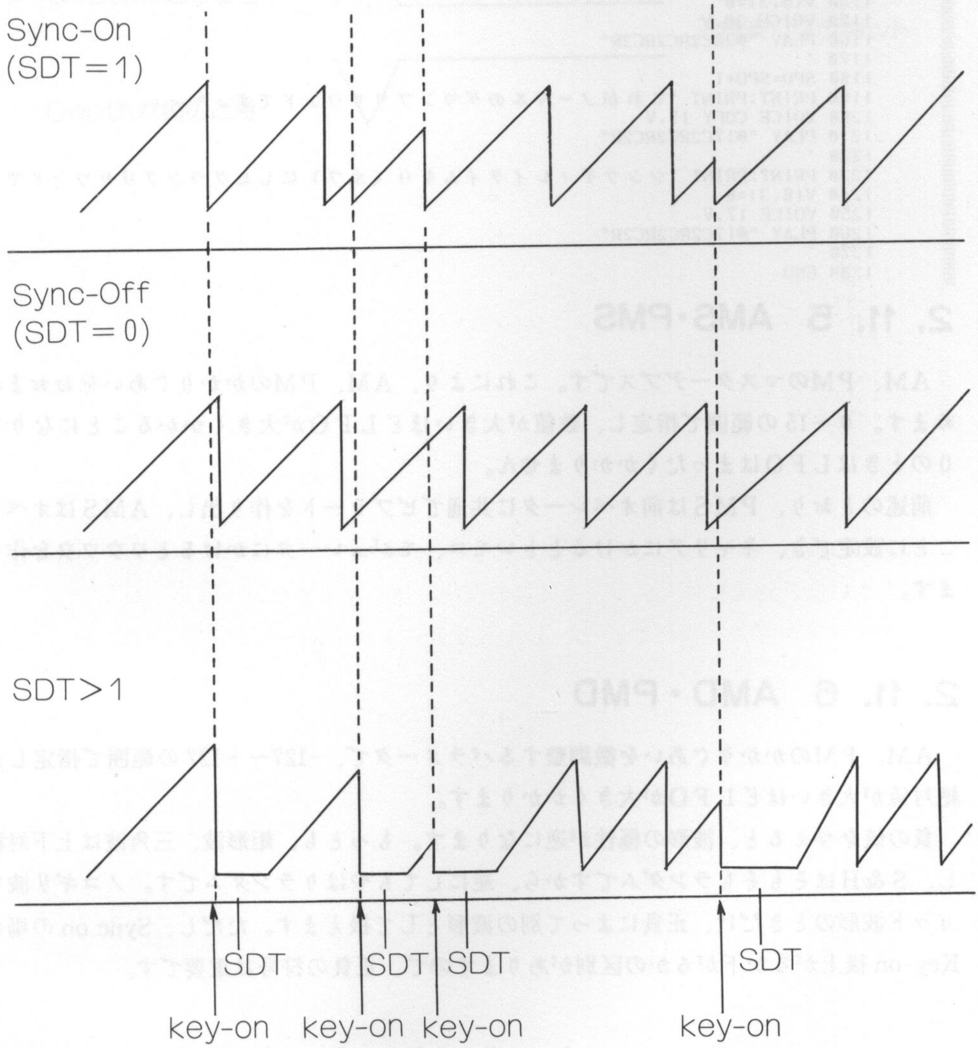
ＬＦＯの速さ、すなわち周波数を決めるパラメータで、０～32767の範囲で指定します。数値が大きいほど周波数は大きく、ＬＦＯの変化は速くなります。

次のサンプルでは、音色番号19－ＬＡＳＥＲ　２　を使ってみました。Speedを尋ねてきますので、適当な値を入力してください。その値にしたがって発音することを繰り返します。プログラムから抜けるにはＳＴＯＰキーを押してください。

```
1000 ' サンプルR　LFO速度
1010 'prog21.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 19,V
1090 SPD=V(0,4)
1100 '
1110 PRINT:PRINT "これがノーマルのレーザーサウンドです。"
1120 PRINT "速度は";SPD
1130 PLAY "@1905C1R"
1140 '
1150 *LOOP
1160   INPUT "速度を入力して下さい: ",SPD
1170   V(0,4)=SPD
1180   VOICE 19,V
1190   PLAY "@19C1R"
1200 GOTO *LOOP
1210 '
1220 END
```

2. 11. 4 Sync Delay-time (SDT)

Key-on（鍵盤を押すことに相当する）とＬＦＯの波形変化とのタイミングを決めるパラメータです。１を指定するとSync onとなり、ＬＦＯの波形はKey-onのたびに中断し、ある一定の値からまた始まります。０を指定するとSync offとなり、ＬＦＯの波形はKey-onと無関係に変化します。２～255の値では、ＬＦＯ効果は指定された時間の経過したのちに現れます。波形の違いを次に示します。



実際に音色にどう影響するのかということですが、楽器音の場合はさしたる影響はありません。効果音の場合は、ＬＦＯを使って音を作っているともいえるので、Sync onにするのが普通です。次のサンプルは、音色番号36－ＦＬＵＴＥと、音色番号17－ＧＲＡＮＰＲＩについて、それぞれSync onの場合とSync offの場合を聞き較べるものです。

```
1000 ' サンプルS　ＬＦＯシンクディレイタイム
1010 'prog22.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 36,V
1090
1100 PRINT:PRINT "これがノーマルのフルートサウンドです。"
1110 PLAY "@36C2RC2RC2R"
1120 '
1130 PRINT:PRINT "シンクディレイタイムを０（オフ）にしたフルートサウンドです。"
1140 V(0,3)=0
1150 VOICE 36,V
1160 PLAY "@36C2RC2RC2R"
1170 '
1180 SPD=SPD+1
1190 PRINT:PRINT "これがノーマルのグランプリサウンドです。"
1200 VOICE COPY 17,V
1210 PLAY "@17C2RC2RC2R"
1220 '
1230 PRINT:PRINT "シンクディレイタイムを０（オフ）にしたグランプリサウンドです。
1240 V(0,3)=0
1250 VOICE 17,V
1260 PLAY "@17C2RC2RC2R"
1270 '
1280 END
```

2. 11. 5 AMS・PMS

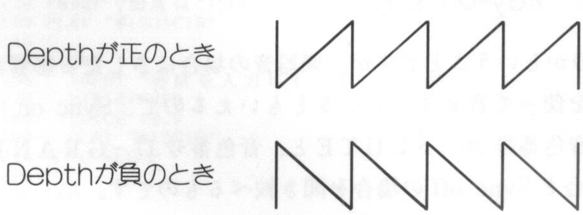
AM、PMのマスターデプスです。これにより、AM、PMのかかりぐあいをおおまかに定めます。0～15の範囲で指定し、数値が大きいほどＬＦＯが大きくかかることになります。0のときはＬＦＯはまったくかかりません。

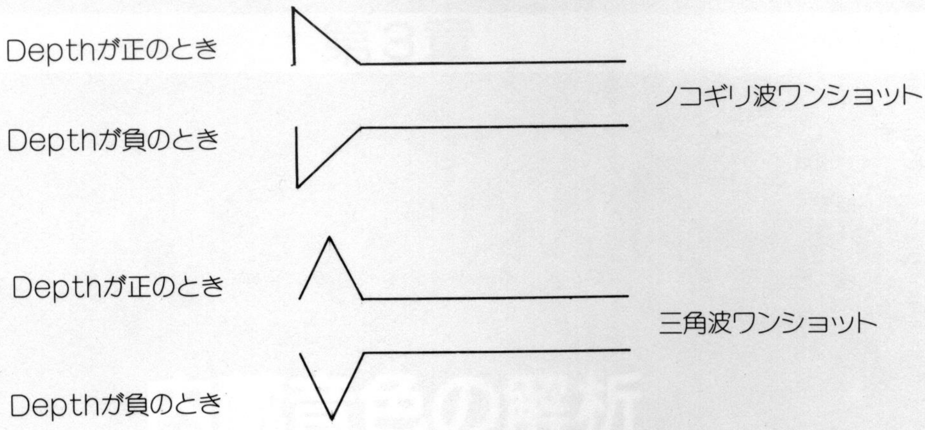
前述のとおり、ＰＭＳは前オペレータに共通でビブラートを作り出し、ＡＭＳはオペレータごとに設定でき、キャリアにかけるとトレモロ、モジュレータにかけるとワウワウを作り出します。

2. 11. 6 AMD・PMD

AM、PMのかかりぐあいを微調整するパラメータで、-127～＋127の範囲で指定します。絶対値が大きいほどＬＦＯが大きくかかります。

負の値を与えると、波形の極性が逆になります。もっとも、矩形波、三角波は上下対称ですし、S&Hはそもそもランダムですから、逆にしてもやはりランダムです。ノコギリ波ワンショット波形のときだけ、正負によって別の波形として扱えます。ただし、Sync on の場合は、Key-on 後上がるか下がるかの区別がありますので、正負の符号は重要です。




Depthが正のとき
Depthが負のとき
ノコギリ波ワンショット
Depthが正のとき
Depthが負のとき
三角波ワンショット

```
1000 ' サンプル5 LFOシンクディレイタイム
1010 ' prog52.bas
1020 CLEAR [illegible]
[illegible]
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,31)
1080 VOICE COPY 30,V
1090 '
1100 PRINT:PRINT "これがノーマルのリードサウンドです。"
1110 PLAY "[illegible]"
1120 '
1130 PRINT:PRINT "シンクディレイタイムを9（オン）にしたリードサウンドです。"
[illegible]
1190 PRINT:PRINT "これがノーマルのグランプリサウンドです。"
1200 VOICE COPY 17,V
1210 PLAY "[illegible]"
1220 '
1230 PRINT:PRINT "シンクディレイタイムを9（オン）にしたグランプリサウンドです。"
[illegible]
1250 VOICE 17,V
1260 PLAY "[illegible]"
1270 '
1280 END
```

2. 11. 5 AMS・PMS

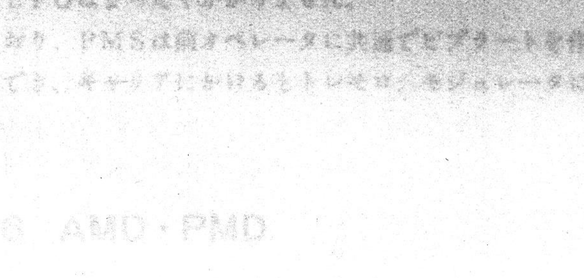
AM、PMのセンシティビティです。これにより、AM、PMのかかり具合いをオペレータごとに設定します。0～3の範囲で設定し、数値が大きいほどLFOが大きくかかることになります。0のときはLFOはまったくかかりません。

前述のとおり、PMSは全オペレータに共通でビブラートを作り出し、AMSはオペレータごとに設定でき、キャリアにかけるとトレモロ、モジュレータにかけるとワウワウを作り出します。

2. 11. 6 AMD・PMD

AM、PMの[illegible]。数値が大きいほどLFOが[illegible]。

負の値を与えると、波形の位相が逆になります。もっとも、三角波、方形波は上下対称ですし、ランダム波はもともとランダムですから、逆にしてもやはりランダムです。ノコギリ波のシンク[illegible]だけ、正負によって別の波形として扱えます。ただし、Sync onの場合はKey-on[illegible]の区別がありますので、正負の符号は重要です。

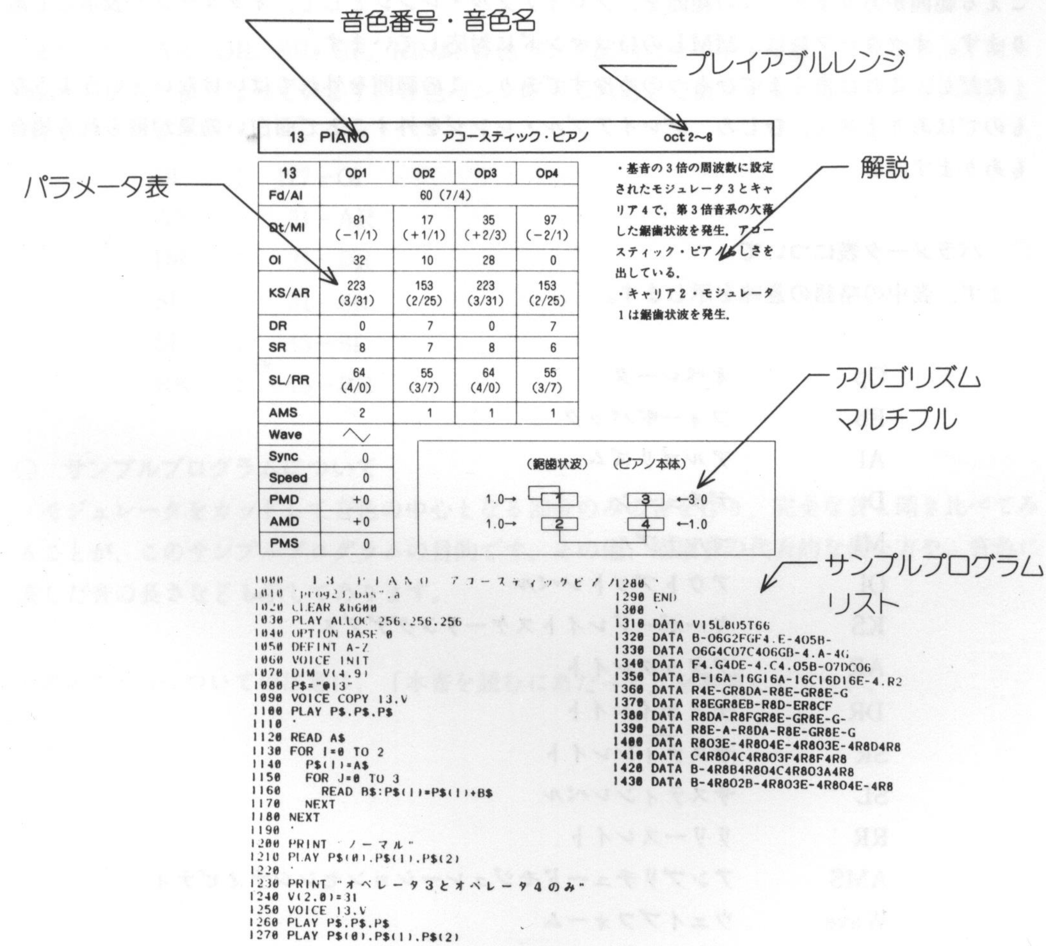


第3章

内蔵音色の解析

ＰＣシリーズには69種類の完成された音色が内蔵されています。そこで、前章までに紹介した音創りのテクニックがこれらの音色にどのように反映されているかを分析することにします。

まず、分析結果の見方を説明します。分析結果は、ひとつの音色に対して１ページを割り当て、豊富な情報を整理された形で見られるように工夫してあります。



13 PIANO	アコースティック・ピアノ	oct 2～8

13	Op1	Op2	Op3	Op4
Fd/Al	60 (7/4)			
Dt/Ml	81 (−1/1)	17 (＋1/1)	35 (＋2/3)	97 (−2/1)
Ol	32	10	28	0
KS/AR	223 (3/31)	153 (2/25)	223 (3/31)	153 (2/25)
DR	0	7	0	7
SR	8	7	8	6
SL/RR	64 (4/0)	55 (3/7)	64 (4/0)	55 (3/7)
AMS	2	1	1	1
Wave	︿﹀			
Sync	0			
Speed	0			
PMD	＋0			
AMD	＋0			
PMS	0			

・基音の３倍の周波数に設定されたモジュレータ３とキャリア４で、第３倍音系の欠落した鋸歯状波を発生。アコースティック・ピアノらしさを出している。
・キャリア２・モジュレータ１は鋸歯状波を発生。

```
1000 ' 13 PIANO アコースティックピアノ
1010 'prog21.bas".a
1020 CLEAR &h600
1030 PLAY ALLOC 256.256.256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4.9)
1080 P$="@13"
1090 VOICE COPY 13.V
1100 PLAY P$.P$.P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル"
1210 PLAY P$(0).P$(1).P$(2)
1220 '
1230 PRINT "オペレータ3とオペレータ4のみ"
1240 V(2.0)=31
1250 VOICE 13.V
1260 PLAY P$.P$.P$
1270 PLAY P$(0).P$(1).P$(2)
1280 '
1290 END
1300 '
1310 DATA V15L8O5T66
1320 DATA B-O6G2FGF4.E-4O5B-
1330 DATA O6G4C07C4O6GB-4.A-4G
1340 DATA F4.G4DE-4.C4.O5B-O7DC06
1350 DATA B-16A-16G16A-16C16D16E-4.R2
1360 DATA R4E-GR8DA-R8E-GR8E-G
1370 DATA R8EGR8EB-R8D-ER8CF
1380 DATA R8DA-R8FGR8E-GR8E-G-
1390 DATA R8E-A-R8DA-R8E-GR8E-G
1400 DATA R8O3E-4R8O4E-4R8O3E-4R8D4R8
1410 DATA C4R8O4C4R8O3F4R8F4R8
1420 DATA B-4R8B4R8O4C4R8O3A4R8
1430 DATA B-4R8O2B-4R8O3E-4R8O4E-4R8
```

○ 音色の番号について

( ) 内は同様の音色が 0 ～12番にも割り当てられていることを示しています。

音色名は、ＰＣシリーズのマニュアルと同じです。また、音色の特徴を一言書き添えました。

○ プレイアブル・レンジについて

楽器には、それぞれ音域がありますから、シンセサイザで合成した音色にも、それらしく聞こえる範囲があります。この範囲を、プレイアブル・レンジとして、オクターブで表示してあります。オクターブ数は、ＭＭＬのＯコマンドに対応しています。

ただし、これはあくまでひとつのめやすであり、この範囲を外れてはいけないというようなものではありません。むしろ、プレイアブル・レンジを外すことで面白い効果が得られる場合もあります。

○ パラメータ表について

まず、表中の略語の意味を示します。

Op	オペレータ
Fb	フィードバック
Al	アルゴリズム
Dt	デチューン
Ml	マルチプル
Ol	アウトプットレベル
KS	キーボードレイトスケーリングデプス
AR	アタックレイト
DR	デイケイレイト
SR	サスティンレイト
SL	サスティンレベル
RR	リリースレイト
AMS	アンプリチュードモジュレーションセンシティビティ
Wave	ウェイブフォーム
Sync	シンクロ
Speed	スピード
PMD	ピッチモジュレーションデプス

AMD	アンプリチュードモジュレーションデプス
PMS	ピッチモジュレーションセンシティビティ

表中のデータで、「／」で仕切られているものは、説明を要します。たとえば、Fb/Al欄に60(7/4)とあるのは、Fb＝７、Al＝４を意味します。60というのは、YM－２２０３のレジスタアドレスＢ０Ｈ～Ｂ２Ｈに書き込めるよう、計算した値です。

また、Ol、AR、DR、SR、SL、RRは、音色バンク配列にセットする値とは異なり、YM－2203に対応した値となっています。音色バンク配列に対応した値とするには、それぞれ次のようにして下さい。

Ol	：	127－Ol
AR	：	31－AR
DR	：	31－DR
SR	：	31－SR
SL	：	15－SL
RR	：	15－RR

○　サンプルプログラムについて

モジュレータをカットして音色の中心となる部分のみの音を作り、完全な音と聞き比べてみることが、このサンプルプログラムの目的です。その他、効果音の代表的な使い方や、音色に適した音の長さなども示してあります。

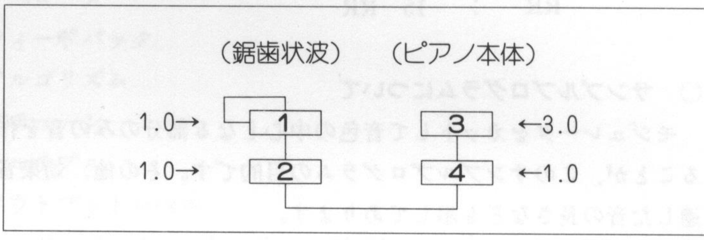
プログラムについての注意は、「本書を読むにあたって」を参照して下さい。

13　PIANO　　アコースティック・ピアノ　　oct 2～8

13	Op1	Op2	Op3	Op4
Fd/Al	60 (7/4)			
Dt/MI	81 (−1/1)	17 (+1/1)	35 (+2/3)	97 (−2/1)
Ol	32	10	28	0
KS/AR	223 (3/31)	153 (2/25)	223 (3/31)	153 (2/25)
DR	0	7	0	7
SR	8	7	8	6
SL/RR	64 (4/0)	55 (3/7)	64 (4/0)	55 (3/7)
AMS	2	1	1	1
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・基音の3倍の周波数に設定されたモジュレータ3とキャリア4で，第3倍音系の欠落した鋸歯状波を発生．アコースティック・ピアノらしさを出している．

・キャリア2・モジュレータ1は鋸歯状波を発生．



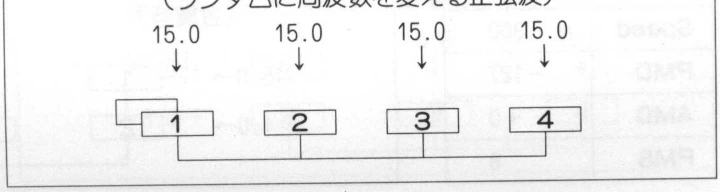

```
1000 ' 1 3  P I A N O  アコースティックピアノ
1010 'prog23.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 P$="@13"
1090 VOICE COPY 13,V
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル"
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ3とオペレータ4のみ"
1240 V(2,0)=31
1250 VOICE 13,V
1260 PLAY P$,P$,P$
1270 PLAY P$(0),P$(1),P$(2)
1280 '
1290 END
1300 '
1310 DATA V15L8O5T66
1320 DATA B-O6G2FGF4.E-4O5B-
1330 DATA O6G4C07C4O6GB-4.A-4G
1340 DATA F4.G4DE-4.C4.O5B-O7DCO6
1350 DATA B-16A-16G16A-16C16D16E-4.R2
1360 DATA R4E-GR8DA-R8E-GR8E-G
1370 DATA R8EGR8EB-R8D-ER8CF
1380 DATA R8DA-R8FGR8E-GR8E-G-
1390 DATA R8E-A-R8DA-R8E-GR8E-G
1400 DATA R8O3E-4R8O4E-4R8O3E-4R8D4R8
1410 DATA C4R8O4C4R8O3F4R8F4R8
1420 DATA B-4R8B4R8O4C4R8O3A4R8
1430 DATA B-4R8O2B-4R8O3E-4R8O4E-4R8
```

14　MUSHI　　風流な虫の声　　oct 5

14	Op1	Op2	Op3	Op4
Fb/Al	7（0/7）			
Dt/ML	63 （+3/15）	95 （−3/15）	95 （−3/15）	63 （+3/15）
Ol	0	0	0	0
KS/AR	30 （0/30）	20 （0/20）	10 （0/10）	2 （0/2）
DR	16	20	20	20
SR	4	20	20	14
SL/RR	167 （10/7）	7 （0/7）	7 （0/7）	7 （0/7）
AMS	0	0	0	2
Wave	S&H			
Sync	1			
Speed	6000			
PMD	13500			
AMD	+127			
PMS	2			

・サンプル&ホールドによるランダムなビブラートをかけた正弦波で消え入るように澄んだ虫の声を実現している.

・4つのキャリアはそれぞれ音量のピークが異なり，複数の虫が鳴いている感じを出している.



（ランダムに周波数を変える正弦波）

```
1000 ' 1 4  M U S H I  風流な虫の声
1010 'prog24.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 READ A$
1080 PLAY A$,A$,A$
1090 '
1100 READ A$,B$,C$
1110 FOR I=0 TO 20
1120   R1=2^(RND(1)*4+2)
1130   R2=2^(RND(1)*4+2)
1140   R3=2^(RND(1)*4+2)
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @14V15L2T66
1210 DATA R=R1;E
1220 DATA R=R2;F
1230 DATA R=R3;F+
```

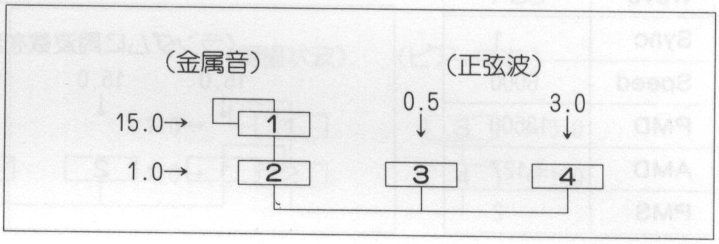
15 DESCENT　　高空からの降下音　　oct 1～8

15	Op1	Op2	Op3	Op4
Fb/Al	62 (7/6)			
Dt/Ml	15 (+0/15)	1 (+0/1)	0 (+0/0)	3 (+0/3)
Ol	37	42	20	0
KS/AR	71 (1/7)	71 (1/7)	71 (1/7)	74 (1/10)
DR	10	10	12	12
SR	2	2	2	2
SL/RR	4 (0/4)	4 (0/4)	3 (0/3)	3 (0/3)
AMS	0	0	0	0
Wave	／｜ (sawtooth)			
Sync	1			
Speed	300			
PMD	−127			
AMD	+0			
PMS	8			

・キャリア2とモジュレータ1は金属音を実現している.

・キャリア3と4は周波数の異なる正弦波で，かなり遅い立ち上がりの音である.

・この音に周期の遅い鋸歯状波でPmdをマイナスにして深いビブラートをかけることにより高い周波数から低い周波数にくり返し降りてくるサウンドを実現している.



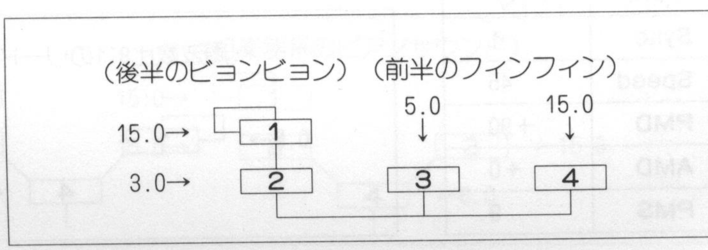

```
1000 ' １ ５  Ｄ Ｅ Ｄ Ｃ Ｅ Ｎ Ｔ  高空からの降下音
1010 'prog25.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 '
1080 READ A$
1090 PLAY A$,A$,A$
1100 READ A$,B$,C$
1110 FOR I=0 TO 9
1120   N1=RND(1)*96
1130   N2=RND(1)*96
1140   N3=RND(1)*96
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @15L1T32
1210 DATA k=n1;
1220 DATA r4k=n2;
1230 DATA r2k=n3;
```

16　UFO　　UFOの飛行音　　oct 3〜7

16	Op1	Op2	Op3	Op4
Fb/Al	54 (6/6)			
Dt/Ml	63 (+3/15)	83 (−3/3)	85 (−3/5)	63 (+3/15)
Ol	15	0	25	15
KS/AR	4 (0/4)	6 (0/6)	14 (0/14)	10 (0/10)
DR	2	8	14	4
SR	2	3	5	10
SL/RR	7 (0/7)	7 (0/7)	7 (0/7)	24 (1/8)
AMS	0	0	0	2
Wave				
Sync	1			
Speed	1000			
PMD	−98			
AMD	+10			
PMS	6			

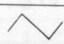
・基本的には，キャリア2とモジュレータ1が超スローアタックのオルガン，キャリア3と4が周波数の異なるスローアタックの正弦波である．

・この音に，中速度・中程度の深さのビブラートを，Pmdがマイナスの鋸歯状波でかけることにより，「フィンフィン」「ビョンビョン」といったUFOの飛行音を実現している．




```
1000 ' 1 6   U F O   U F Oの飛行音
1010 'prog26.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 '
1080 READ A$
1090 PLAY A$,A$,A$
1100 READ A$,B$,C$
1110 FOR I=0 TO 9
1120   N1=RND(1)*96
1130   N2=RND(1)*96
1140   N3=RND(1)*96
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @1611t32
1210 DATA k=n2;
1220 DATA r4k=n2;
1230 DATA r2k=n3;
```

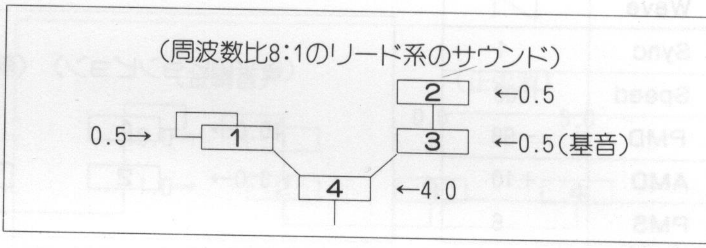
17　GRANPRI　　レーシングカーのエキゾースト　　oct 4～5

17	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	0 (+0/0)	48 (+3/0)	48 (+3/0)	52 (+3/4)
Ol	14	20	6	0
KS/AR	76 (1/12)	80 (1/16)	157 (2/29)	137 (2/9)
DR	5	4	6	4
SR	1	1	5	31
SL/RR	85 (5/5)	37 (2/5)	37 (2/5)	168 (10/8)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	45			
PMD	+90			
AMD	+0			
PMS	9			

・LFOのスピードを遅くしてピッチEGとして用いたセッティング．

・PmdプラスSync ONだと発音と同時に周波数が上昇していく．下降に転じる付近で音が減衰してしまうので上昇音だけが聞こえる．

・LFO効果を除くとリードオルガンのような音である．




```
1000 ' 1 7  G R A N D P R I X  レーシングカーのエキゾースト
1010 'prog27.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 '
1080 READ A$
1090 PLAY A$,A$,A$
1100 READ A$,B$,C$
1110 FOR I=0 TO 20
1120   R1=2^(RND(1)*4+2)
1130   R2=2^(RND(1)*4+2)
1140   R3=2^(RND(1)*4+2)
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @1712t50
1210 DATA r=r1;e
1220 DATA r=r2;f
1230 DATA r=r3;f+
```

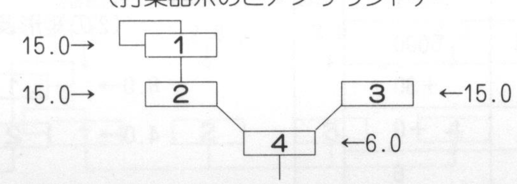
18　LASER　1　　レーザーガンの発射音（短）　　oct 2～8

18	Op1	Op2	Op3	Op4
Fb/Al	35（4/3）			
Dt/Ml	63 （+3/15）	95 （−3/15）	95 （−3/15）	54 （+3/6）
Ol	49	44	17	0
KS/AR	31 （0/31）	31 （0/31）	31 （0/31）	159 （2/31）
DR	26	26	17	12
SR	5	9	1	1
SL/RR	11 （0/11）	11 （0/11）	167 （10/7）	248 （15/8）
AMS	0	0	0	0
Wave	(waveform)			
Sync	1			
Speed	16000			
PMD	−127			
AMD	+0			
PMS	7			

・キャリア4とモジュレータ1，2，3との周波数比が1：2.5である．これだけならば，打楽器としてのピアノサウンドを得るセッティングである．

・ただし，非常に速くて深いビブラートとデチューン効果により音程感を失くし，レーザーガンの短い発射音を実現している．

（打楽器系のピアノサウンド）



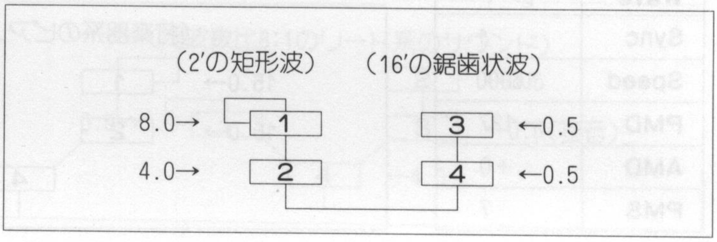

```
1000 ' 1 8  L A S E R 1  レーザーガンの発射音
1010 'prog28.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 '
1080 READ A$
1090 PLAY A$,A$,A$
1100 READ A$,B$,C$
1110 FOR I=0 TO 20
1120   R1=2^(RND(1)*4+2)
1130   R2=2^(RND(1)*4+2)
1140   R3=2^(RND(1)*4+2)
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @18l16o5t66
1210 DATA r=r1;eeeeeeee
1220 DATA r=r2;ffffffff
1230 DATA r=r3;gggggggg
```

19　LASER　2　　レーザーガンの発射音（長）　　oct 5〜6

19	Op1	Op2	Op3	Op4
Fb/Al	4（0/4）			
Dt/Ml	88（−3/8）	84（−3/4）	96（−2/0）	16（+1/0）
Ol	30	10	37	0
KS/AR	146（2/18）	146（2/18）	146（2/18）	146（2/18）
DR	2	2	2	2
SR	0	0	0	0
SL/RR	7（0/7）	7（0/7）	5（0/5）	6（0/6）
AMS	0	0	0	0
Wave				
Sync	1			
Speed	5000			
PMD	+30			
AMD	+0			
PMS	8			

・キャリア2とモジュレータ1は2'の矩形波によるオルガンサウンド，キャリア4とモジュレータ3は16'の鋸歯状波によるオルガンサウンドである．

・この音にデチューン効果と三角波によるビブラートをかけることにより持続的なレーザーガンの発射音を実現している．




```
1000 ' 1 9  L A S E R 2  レーザーガンの発射音
1010 'prog29.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 '
1080 READ A$
1090 PLAY A$,A$,A$
1100 READ A$,B$,C$
1110 FOR I=0 TO 20
1120   R1=2^(RND(1)*4+2)
1130   R2=2^(RND(1)*4+2)
1140   R3=2^(RND(1)*4+2)
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @19v15l1o5t66
1210 DATA r=r1;e
1220 DATA r=r2;f
1230 DATA r=r3;f+
```

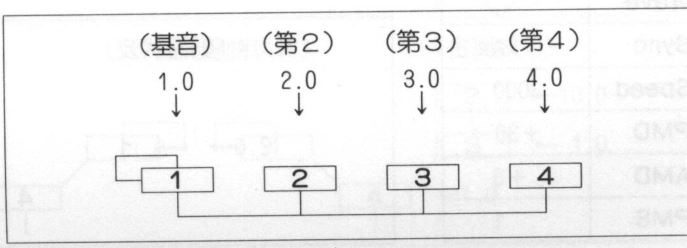
20　SIN　WAVE　　チューニング用正弦波　　oct 4〜8

20	Op1	Op2	Op3	Op4
Fb/Al	7 (0/7)			
Dt/MI	1 (+0/1)	2 (+0/2)	3 (+0/3)	4 (+0/4)
Ol	0	0	0	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	0	0	0	0
SR	0	0	0	0
SL/RR	31 (1/15)	31 (1/15)	31 (1/15)	31 (1/15)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・音色の時間的な変化がないので他の楽器とのチューニングに使える．

・この音色が澄んだ音色に聞こえない場合は，オーディオアンプのボリュームを下げること．

・音色名はSIN WAVEだが，実際は基音〜第4倍音までの正弦波の合成音である．




```
1000 ' 2 0  S I N E  W A V E  チューニング用正弦波
1010 'prog30.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 '
1080 PLAY "@20","@20","@20"
1090 '
1100 PRINT "楽器を調整し、出力レベルを設定します。"
1110 PRINT "キーを押してください（音がオン／オフします）。"
1120 '
1130 VOICE REG &HA4,&H2C
1140 VOICE REG &HA0,&H81
1150 '
1160 IF INKEY$="" THEN 1160
1170 KE=&HF0-KE
1180 VOICE REG &H28,KE
1190 GOTO 1160
```

21　BRASS　1　ホルン系のブラスサウンド　　oct 2〜4

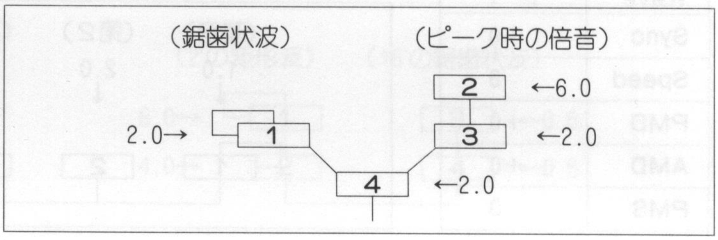
21	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	2 (+0/2)	38 (+2/6)	2 (+0/2)	18 (+1/2)
Ol	25	32	42	0
KS/AR	141 (2/13)	79 (1/15)	21 (0/21)	82 (1/18)
DR	6	8	7	4
SR	2	0	0	0
SL/RR	24 (1/8)	24 (1/8)	40 (2/8)	40 (2/8)
AMS	0	0	0	0
Wave				
Sync	6			
Speed	3000			
PMD	+30			
AMD	+0			
PMS	1			

・モジュレータ1と3が，キャリア4を変調して鋸歯状波を発生している．

・基音が2.0なので実音はオクターブ上になる．

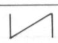
・モジュレータ2で音量ピーク時付近の高次倍音を補っている．

・ビブラートがかけてあるが，なくても十分である．

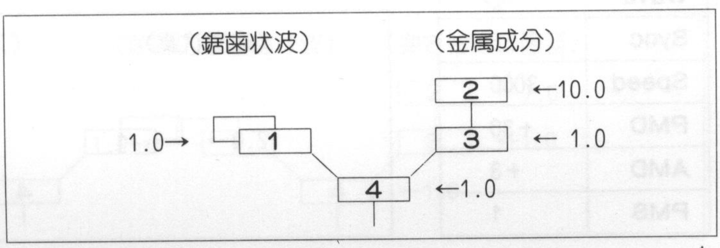



```
1000 ' ２１　ＢＲＡＳＳ１　ホルン系のブラスサウンド
1010 'prog31.bas",a
1020 CLEAR &H600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 21,V
1090 '
1100 P$="@21"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 1
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ１，４のみ
1250   V(3,0)=31
1260   VOICE 21,V
1270 PLAY P$(0),P$(1),P$(2)
1280 END
1290 DATA v15l4o3t66
1300 DATA f+2b4.o4c+8c+2o3b2
1310 DATA o4f+ag+ef+1r2
1320 DATA f+2f+4.f+8f+2f+2
1330 DATA o4ddddd1
1340 DATA d2d4.d8d2d2
1350 DATA bbbbb1
```

22(1) BRASS　低音域の金管サウンド　oct 2〜3

22(1)	Op1	Op2	Oo3	Op4
Fb/Al	58 (7/2)			
Dt/MI	1 (+0/1)	42 (+2/10)	113 (−1/1)	1 (+0/1)
Ol	26	30	52	0
KS/AR	142 (2/14)	89 (1/25)	79 (1/15)	82 (1/18)
DR	7	8	8	3
SR	0	3	0	0
SL/RR	53 (3/5)	175 (10/15)	53 (3/5)	40 (2/8)
AMS	1	1	1	1
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・モジュレータ1と3が，キャリア4を変調して鋸歯状波を発生している．
・ただし，アタックが速く減衰形のエンベロープを持つモジュレータ2により，吹き始めに金属的な成分を加えて，割れたようなブラスを実現している．




```
1000 ' 2 2 ( 1 )   B R A S S  2  低音域の金管サウンド
1010 'prog32.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 22,V
1090 '
1100 P$="@22"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 1
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ2，3，4のみ
1250 V(1,0)=31
1260 VOICE 22,V
1270 PLAY P$(0),P$(1),P$(2)
1280 END
1290 DATA v15l4o3t66
1300 DATA f+2b4.o4c+8c+2o3b2
1310 DATA o4f+ag+ef+1r2
1320 DATA f+2f+4.f+8f+2f+2
1330 DATA o4ddddd1
1340 DATA d2d4.d8d2d2
1350 DATA bbbbb1
```

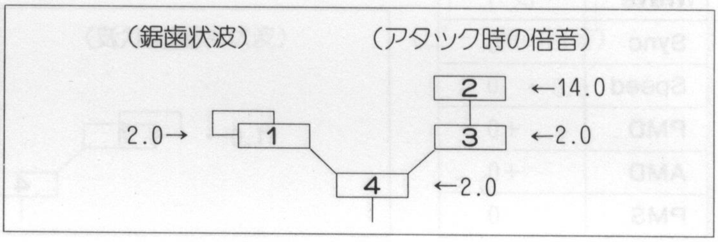
23　TRUMPET　　明るい響きのトランペット　　oct 5

23	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	2 (+0/2)	14 (+0/14)	2 (+0/2)	2 (+0/2)
Ol	23	40	38	0
KS/AR	140 (2/12)	140 (2/12)	140 (2/12)	83 (1/19)
DR	13	13	13	3
SR	0	0	0	0
SL/RR	19 (1/3)	250 (15/10)	19 (1/3)	10 (0/10)
AMS	0	0	0	0
Wave				
Sync	6			
Speed	3000			
PMD	+30			
AMD	+3			
PMS	1			

・モジュレータ1と3が，キャリア4を変調して鋸歯状波を発生している．

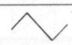
・減衰が速いモジュレータ2で，アタック時の高次倍音を補っている．

・ビブラートがかけてあるが，なくても十分である．

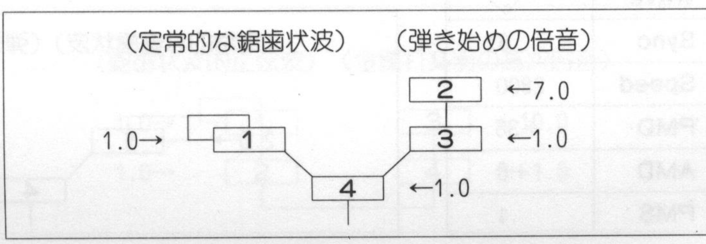



```
1000 ' ２３　ＴＲＵＭＰＥＴ　明るい響きのトランペット
1010 'prog33.bas".a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 23,V
1090 '
1100 P$="@23q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 1
1150   P$(I)=A$
1160   FOR J=0 TO 1
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1)
1230 '
1240 PRINT "オペレータ２，３，４のみ
1250   V(1,0)=31
1260   VOICE 23.V
1270 PLAY P$(0),P$(1),P$(2)
1280 END
1290 DATA v15l4o3t66
1300 DATA f+2b4.o4c+8c+2o3b2
1310 DATA o4f+ag+ef+1r2
1320 DATA f+2f+4.f+8f+2f+2
1330 DATA o4ddddd1
1340 DATA d2d4.d8d2d2
1350 DATA bbbbb1
```

24　STRING　1　重厚なストリングス　oct 5～7

24	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	17 (+1/1)	23 (+1/7)	1 (+0/1)	17 (+1/1)
Ol	28	30	40	0
KS/AR	84 (1/20)	89 (1/25)	84 (1/20)	78 (1/14)
DR	10	10	13	4
SR	0	0	0	0
SL/RR	21 (1/5)	88 (5/8)	38 (2/6)	6 (0/6)
AMS	0	0	0	0
Wave				
Sync	8			
Speed	3000			
PMD	+40			
AMD	+3			
PMS	1			

・モジュレータ1と3が，キャリア4を変調して鋸歯状波を発生している．
・ブラス系の音色とセッティングが似ているが，モジュレータのアタックが速くディケイが遅いので，倍音構成は定常的になる．
・モジュレータ2は弾き始めの高次倍音を補っている．
・ビブラートがかけてあるが，なくても十分である．




```
1000 ' ２４　 ＳＴＲＩＮＧ１　重厚なストリングス
1010 'prog34.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 24,V
1090 P$="@24"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 2
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル"
1210 PLAY P$(0),P$(1),P$(2)
1220 PRINT "オペレータ２，３，４のみ"
1230   V(1,0)=31
1240   VOICE 24,V
1250   PLAY P$(0),P$(1),P$(2)
1260 END
1270 '
1280 DATA v15l8o5t132
1290 DATA r8ab-o6dco5b-16a16gcfgab-16a16g2
1300 DATA r2r8g16a16b-gar4b-gg16a16b-g
1310 DATA a4o6c4o5b-4.ao6d4c4o5gg16a16b-gr2
1320 DATA r1fefg16f16e2
1330 DATA ro4ab-o5dc2r8o4ab-o5dc2
1340 DATA c4f4e4.fb-4a4e2
1350 DATA c1.c2.
1360 DATA f4d4e4.cf4d4e4.c
1370 DATA f4o6c4c4.ff4c4c2
```

25(2)　STRING　2 ストリングスキーボードのサウンド　oct 4〜6

25(2)	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	18 (+1/2)	26 (+1/10)	2 (+0/2)	18 (+1/2)
Ol	31	24	42	0
KS/AR	89 (1/25)	83 (1/19)	89 (1/25)	77 (1/13)
DR	7	9	7	4
SR	0	0	0	0
SL/RR	19 (1/3)	50 (3/2)	19 (1/3)	7 (0/7)
AMS	0	0	0	0
Wave				
Sync	6			
Speed	2800			
PMD	+36			
AMD	+0			
PMS	1			

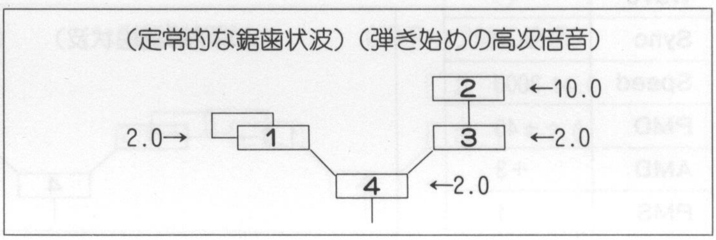
・モジュレータ1と3が，キャリア4を変調して鋸歯状波を発生している．

・モジュレータのアタックが速くディケイが遅いので倍音構成は定常的になる．

・基音が2.0なので実音はオクターブ上になる．

・モジュレータ2は弾き始めの高次倍音を補っている．

・ビブラートがないと少し物足りない感じがする．




```
1000 ' 25 (2)   STRING2  ストリングキーボードのサウンド
1010 'prog35.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 25,V
1090 P$="@25"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 2
1160     READ B$:P$(I)=P$(I)+B$
1170  NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル"
1210 PLAY P$(0),P$(1),P$(2)
1220 PRINT "オペレータ2，3，4のみ"
1230   V(1,0)=31
1240   VOICE 25,V
1250   PLAY P$(0),P$(1),P$(2)
1260 END
1270 '
1280 DATA v15l8o5t132
1290 DATA r8ab-o6dco5b-16a16gcfgab-16a16g2
1300 DATA r2r8g16a16b-gar4b-gg16a16b-g
1310 DATA a4o6c4o5b-4.ao6d4c4o5gg16a16b-gr2
1320 DATA r1fefg16f16e2
1330 DATA ro4ab-o5dc2r8o4ab-o5dc2
1340 DATA c4f4e4.fb-4a4e2
1350 DATA c1.c2.
1360 DATA f4d4e4.cf4d4e4.c
1370 DATA f4o6c4c4.ff4c4c2
```

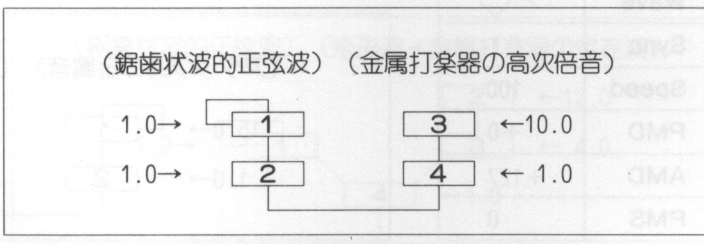
26　EPIANO　1　　代表的な電気ピアノ　　oct 4～5

26	Op1	Op2	Op3	Op4
Fb/Al	20（2/4）			
Dt/Ml	17 （＋1/1）	33 （＋2/1）	10 （＋0/10）	17 （＋1/1）
Ol	25	10	57	0
KS/AR	154 （2/26）	152 （2/24）	218 （3/26）	216 （3/24）
DR	15	12	7	12
SR	0	5	3	5
SL/RR	38 （2/6）	40 （2/8）	70 （4/6）	40 （2/8）
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	＋0			
AMD	＋0			
PMS	0			

・金属片を叩いて音を出すタイプの電気ピアノの音色は本質的には正弦波である．

・キャリア2とオペレータ1で鋸歯状波的な正弦波を発生している．

・モジュレータ3がキャリア4を変調して第9，第11倍音を発生．金属を叩いた時特有の倍音を合成している．




```
1000 '２６　ＥＰＩＡＮＯ１　代表的な電気ピアノ
1010 'prog36.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 26,V
1090 '
1100 P$="@26"
1110 PLAY P$,P$,P$
1120 R=&H88:D=&H8A
1130 READ A$
1140 FOR I=0 TO 2
1150    P$(I)=A$
1160    FOR J=0 TO 3
1170      READ B$:P$(I)=P$(I)+B$
1180    NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル"
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ３，４のみ"
1250    V(2,0)=31
1260    VOICE 26,V
1270    PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t66
1310 DATA b-o6g2fgf4.e-4o5b
1320 DATA o6g4co7c4o6gb-4.a4g
1330 DATA f4.g4de-4.c4.o5b-o7dco6
1340 DATA b-16a-16g16a-16c16d16e-4.r2
1350 DATA r4e-gr8da-r8e-gr8e-g
1360 DATA r8egr8eb-r8d-er8cf
1370 DATA r8da-r8fgr8e-gr8e-g-
1380 DATA r8e-a-r8da-r8e-gr8e-g
1390 DATA r8o3e-4r8o4e-4r8o3e-4r8d4r8
1400 DATA c4r8o4c4r8o3f4r8f4r8
1410 DATA b-4r8b4r8o4c4r8o3a4r8
1420 DATA b-4r8o2b-4r8o3e-4r8o4e-4r8
```

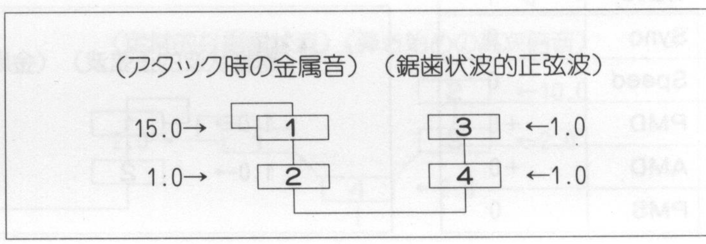
27 EPIANO 2 ダーティなエレピサウンド oct 5〜7

27	Op1	Op2	Op3	Op4
Fb/Al	28 (3/4)			
Dt/Ml	63 (+3/15)	1 (+0/1)	1 (+0/1)	1 (+0/1)
Ol	35	27	42	0
KS/AR	159 (2/31)	219 (3/27)	158 (2/30)	94 (1/30)
DR	15	7	6	6
SR	8	10	11	0
SL/RR	138 (8/10)	134 (8/6)	246 (15/6)	247 (15/7)
AMS	1	3	1	15
Wave				
Sync	1			
Speed	100			
PMD	+0			
AMD	+127			
PMS	0			

・キャリア4とモジュレータ3は鋸歯状波的正弦波であり，リードの振動を実現している．

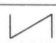
・キャリア2とモジュレータ1は高次倍音を作り出しているが，減衰が速く，アタック時にしか聞こえない．

・トレモロ等がかけてあるが，なくても十分である．




```
1000 '２７　ＥＰＩＡＮＯ２　ダーティなエレピサウンド
1010 'prog37.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 27,V
1090 '
1100 P$="@27"
1110 PLAY P$,P$,P$
1120 R=&H88:D=&H8A
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル"
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ１，２のみ"
1250   V(4,0)=31
1260   VOICE 27,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t66
1310 DATA b-o6g2fgf4.e-4o5b
1320 DATA o6g4co7c4o6gb-4.a-4g
1330 DATA f4.g4de-4.c4.o5b-o7dco6
1340 DATA b-16a-16g16a-16c16d16e-4.r2
1350 DATA r4e-gr8da-r8e-gr8e-g
1360 DATA r8egr8eb-r8d-er8cf
1370 DATA r8da-r8fgr8e-gr8e-g-
1380 DATA r8e-a-r8da-r8e-gr8e-g
1390 DATA r8o3e-4r8o4e-4r8o3e-4r8d4r8
1400 DATA c4r8o4c4r8o3f4r8f4r8
1410 DATA b-4r8b4r8o4c4r8o3a4r8
1420 DATA b-4r8o2b-4r8o3e-4r8o4e-4r8
```

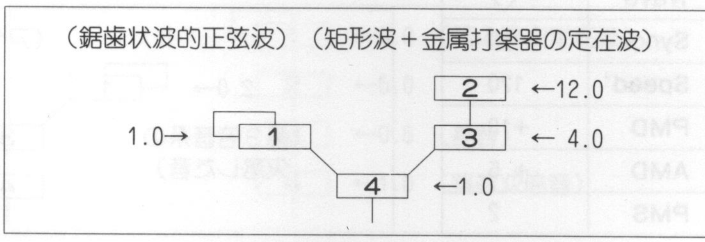
28(3)　EPIANO　3 ライトな感覚のエレピ　　oct 3〜5

28(3)	Op1	Op2	Op3	Op4
Fb/AI	58 (7/2)			
Dt/MI	33 (+2/1)	12 (+0/12)	116 (−1/4)	1 (+0/1)
OI	33	71	45	0
KS/AR	156 (2/28)	219 (3/27)	156 (2/28)	218 (3/26)
DR	4	9	4	3
SR	3	1	3	0
SL/RR	23 (1/7)	2 (0/2)	6 (0/6)	165 (10/5)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・キャリア4とモジュレータ1で鋸歯状波的正弦波を発生.

・同時にキャリア4とモジュレータ3で第2倍音の欠けた矩形波的なライトなサウンドを実現している.

・さらにモジュレータ2の変調で金属打楽器特有の定在波を発生させている.




```
1000 '２８（３）　ＥＰＩＡＮＯ３　ライトな感覚のエレピサウンド
1010 'prog38.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 28,V
1090 '
1100 P$="@28"
1110 PLAY P$,P$,P$
1120 R=&H88:D=&H8A
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル"
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ２，３，４のみ"
1250   V(1,0)=31
1260   VOICE 28,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t66
1310 DATA b-o6g2fgf4.e-4o5b
1320 DATA o6g4co7c4o6gb-4.a-4g-
1330 DATA f4.g4de-4.c4.o5b-o7dco6
1340 DATA b-16a-16g16a-16c16d16e-4.r2
1350 DATA r4e-gr8da-r8e-gr8e-g
1360 DATA r8egr8eb-r8d-er8cf
1370 DATA r8da-r8fgr8e-gr8e-g-
1380 DATA r8e-a-r8da-r8e-gr8e-g
1390 DATA r8o3e-4r8o4e-4r8o3e-4r8d4r8
1400 DATA c4r8o4c4r8o3f4r8f4r8
1410 DATA b-4r8b4r8o4c4r8o3a4r8
1420 DATA b-4r8o2b-4r8o3e-4r8o4e-4r8
```

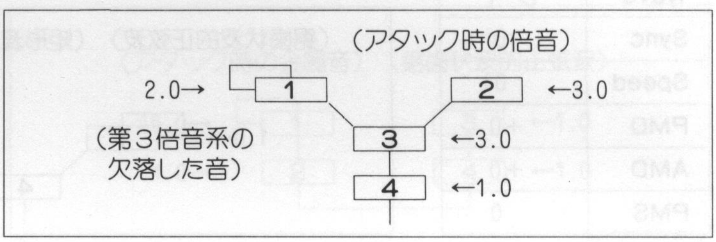
29 GUITAR　ガット・ギターのサウンド　oct 4～6

29	Op1	Op2	Op3	Op4
Fb/Al	57 (7/1)			
Dt/Ml	50 (+3/2)	51 (+3/3)	51 (+3/3)	49 (+3/1)
Ol	40	26	40	0
KS/AR	223 (3/31)	223 (3/31)	223 (3/31)	159 (2/31)
DR	9	6	6	12
SR	4	4	4	3
SL/RR	247 (15/7)	251 (15/11)	251 (15/11)	11 (0/11)
AMS	1	1	1	10
Wave				
Sync	20			
Speed	120			
PMD	+10			
AMD	+ 5			
PMS	2			

・キャリア4と基音の3倍の周波数をもつモジュレータ3で，第3倍音の系列が欠落した撥弦系弦楽器の音を実現している．

・アタックが速い減衰形のエンベロープを持つモジュレータ1と2は，アタック時の倍音を補っている．

・ビブラートとワウがかけてあるが浅いので，なくても十分である．




```
1000 ' 2 9  G U I T A R  ガットギターのサウンド
1010 'prog39.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 29,V
1090 '
1100 P$="@29q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ3, 4のみ
1250   V(1,0)=31
1260   V(2,0)=31
1270   VOICE 29,V
1280   PLAY P$(0),P$(1).P$(2)
1290 END
1300 '
1310 DATA v15l8o5t132
1320 DATA o6e2d4c2d4
1330 DATA e1r4f4
1340 DATA g2f4e2f4
1350 DATA g1.r2
1360 DATA r8eo6co5eber8eo6co5ebe
1370 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1380 DATA r8go6eo5go6do5gr8go6co5go6do5g
1390 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1400 DATA o4a2.a2.
1410 DATA a2.a2.
1420 DATA o5c2.c2.
1430 DATA c2.c2.
```

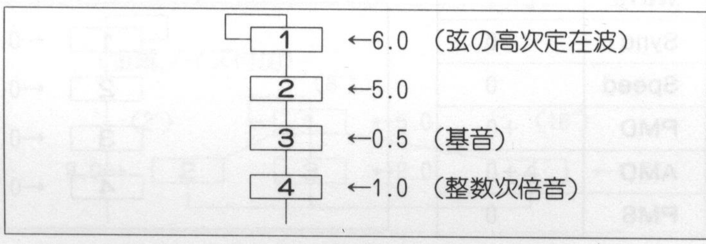
30(4)　EBASS　1　チョッパー風エレクトリックベース　oct 3〜4

30(4)	Op1	Op2	Op3	Op4
Fb/Al	32 (4/0)			
Dt/Ml	54 (+3/6)	53 (+3/5)	48 (+3/0)	49 (+3/1)
Ol	28	58	22	0
KS/AR	223 (3/31)	223 (3/31)	159 (2/31)	159 (2/31)
DR	7	6	9	6
SR	7	6	6	8
SL/RR	41 (2/9)	25 (1/9)	25 (1/9)	249 (15/9)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・基音はモジュレータ３の0.5であり，実音は１オクターブ下となっている．

・弦の振動による整数次の倍音はキャリア４とモジュレータ３で発生している．

・また，モジュレータ１，２の変調により，チョッパー奏法時の高次倍音を発生させている．



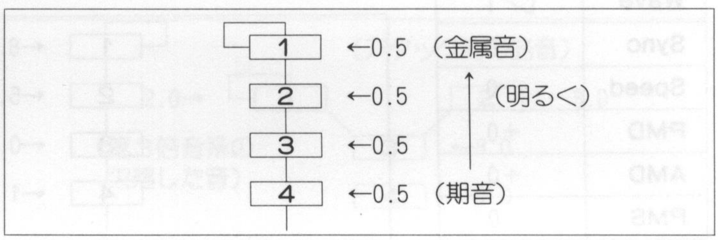

```
1000 ' ３０（４）　ＥＢＡＳＳ１　チョッパー風エレクトリックベース
1010 'prog40.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 30,V
1090 '
1100 P$="@30q7"
1110 PLAY P$,P$,P$
1120 R=&H88:D=&H8A
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ３，４のみ
1250   V(2,0)=31
1260   VOICE 30,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

31　EBASS　2　代表的なエレクトリックベース　oct 3〜4

31	Op1	Op2	Op3	Op4
Fb/AL	48 (6/0)			
Dt/Ml	48 (+3/0)	48 (+3/0)	48 (+3/0)	48 (+3/0)
Ol	23	63	23	0
KS/AR	158 (2/30)	216 (3/24)	220 (3/28)	220 (3/28)
DR	14	10	4	5
SR	6	6	6	6
SL/RR	182 (11/6)	182 (11/6)	182 (11/6)	182 (11/6)
AMS	0	0	0	0
Wave	(鋸歯状波)			
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・キャリア4と同じマルチプルのモジュレータを積み重ねている．モジュレータの積み重ねが増えるほど，明るい金属的な鋸歯状波を得ることができる．

・基音は0.5なので実音は1オクターブ下となる．



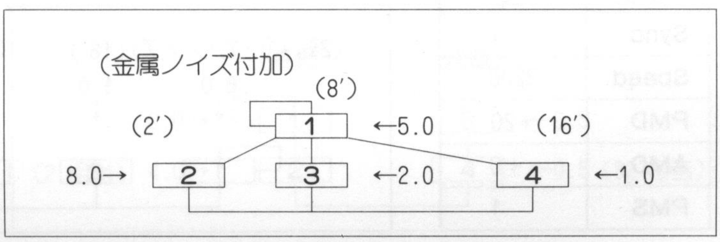

```
1000 ' 3 1  E B A S S 2  代表的なエレクトリックベース
1010 'prog41.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 31,V
1090 '
1100 P$="@31q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ3，4のみ
1250   V(2,0)=31
1260   VOICE 31,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

32(5)　EORGAN　1　代表的なエレクトリックオルガン　oct 3〜8

32(5)	Op1	Op2	Op3	Op4
Fb/Al	61 (7/5)			
Dt/Ml	53 (+3/5)	120 (−1/8)	98 (−2/2)	49 (+3/1)
Ol	42	17	27	5
KS/AR	223 (3/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	18	4	15	15
SR	0	0	0	0
SL/RR	47 (2/15)	15 (0/15)	15 (0/15)	15 (0/15)
AMS	0	0	0	0
Wave	/\/			
Sync	1			
Speed	2800			
PMD	+8			
AMD	+0			
PMS	5			

・キャリア4が16′，キャリア3が8′，キャリア2が2′に相当し，第3倍音のないストレートなエレクトリックオルガンサウンドを実現している．

・3つのキャリアを変調しているモジュレータ1は金属的な共振音を付加し，オーバードライブ効果を出している．




```
1000 ' 32 (5)    ORGAN1  代表的なエレクトリックオルガン
1010 'prog42.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 32,V
1090 '
1100 P$="@32"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ3，4のみ
1240   V(2,0)=31
1250   VOICE 32,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l16o5t66
1300 DATA o4g4o5d4o4b-4.a8g8b-8a8g8
1310 DATA f+8a8d4g8d8a8d8
1320 DATA b-8aga8d8g8dga8dab-8agado5dc
1330 DATA o4b-agb-agf+agdgab-o5cder2
1340 DATA r2r8d8g8a8b-o6co5b-o6cc8.o5b-32o6c32
1350 DATA dcde-dco5b-ab-o6dcdo5f+o6dcd
1360 DATA o5go6dcdo5f+o6dcdo5b-4o6c4d4c4
1370 DATA r8o5b-8r8a8r8d8gab-g
1380 DATA r2r1
1390 DATA r2d8r8c8r8
1400 DATA d8r8c8r8.gf+gr16gf+gr16gf+g
1410 DATA d4c4o4b-4
```

33 EORGAN 2 16′,·8′, 5⅓′に2⅔′のパーカッシブ oct 4〜8

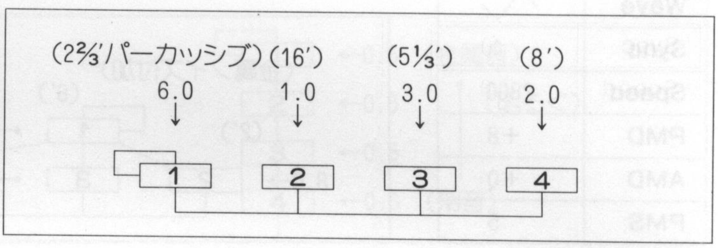
33	Op1	Op2	Op3	Op4
Fb/Al	31 (3/7)			
Dt/Ml	102 (−2/6)	49 (+3/1)	83 (−3/3)	34 (+2/2)
Ol	12	10	10	10
KS/AR	28 (0/28)	159 (2/31)	31 (0/31)	31 (0/31)
DR	18	15	15	15
SR	0	0	0	0
SL/RR	255 (15/15)	15 (0/15)	15 (0/15)	15 (0/15)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	3200			
PMD	+20			
AMD	+0			
PMS	1			

・サイン波合成のセッティングの好例である．

・キャリア2が16′，キャリア3が5⅓′，キャリア4が4′に相当する正弦波である．

・キャリア1は減衰形のエンベロープを持ち2⅔′のパーカッシブ・カプラーとなる．

・ビブラートがかけてあるが，デチューンで得られるレスリー効果だけでも十分である．




```
1000 ' 3 3  E O R G A N 2  16',8',5(1/3)'に2(2/3)のパーカッシブ
1010 'prog43.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 33,V
1090 '
1100 P$="@33"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ2，3，4のみ
1240   V(1,0)=31
1250   VOICE 33,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l16o5t66
1300 DATA o4g4o5d4o4b-4.a8g8b-8a8g8
1310 DATA f+8a8d4g8d8a8d8
1320 DATA b-8aga8d8g8dga8dab-8agado5dc
1330 DATA o4b-agb-agf+agdgab-o5cder2
1340 DATA r2r8d8g8a8b-o6co5b-o6cc8.o5b-32o6c32
1350 DATA dcde-dco5b-ab-o6dcdo5f+o6dcd
1360 DATA o5go6dcdo5f+o6dcdo5b-4o6c4d4c4
1370 DATA r8o5b-8r8a8r8d8gab-g
1380 DATA r2r1
1390 DATA r2d8r8c8r8
1400 DATA d8r8c8r8.gf+gr16gf+gr16gf+g
1410 DATA d4c4o4b-4
```

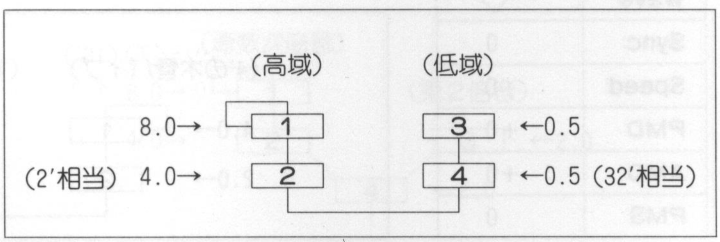
34(6)　PORGAN　1 代表的なパイプオルガン　oct 4〜7

34(6)	Op1	Op2	Op3	Op4
Fb/Al	4 (0/4)			
Dt/Ml	88 (−3/8)	84 (−3/4)	96 (−2/0)	16 (+1/0)
Ol	31	27	31	0
KS/AR	148 (2/20)	144 (2/16)	146 (2/18)	147 (2/19)
DR	31	31	31	31
SR	0	0	0	0
SL/RR	7 (0/7)	7 (0/7)	5 (0/5)	8 (0/8)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・基音は0.5なので，キャリア4，モジュレータ3が16′，キャリア2が2′，モジュレータ1が1′に相当する．

・レイト・スケーリングのかけられたキャリア4とモジュレータ3で低域の金管パイプを実現している．

・高域はキャリア2とモジュレータ1で発生させている．




```
1000 ' 3 4 ( 6 )   P O R G A N 1  代表的なパイプオルガン
1010 'prog44.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 34,V
1090 '
1100 P$="@34"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ3, 4のみ
1240   V(2,0)=31
1250   VOICE 34,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l16o5t66
1300 DATA o4g4o5d4o4b-4.a8g8b-8a8g8
1310 DATA f+8a8d4g8d8a8d8
1320 DATA b-8aga8d8g8dga8dab-8agado5dc
1330 DATA o4b-agb-agf+agdgab-o5cder2
1340 DATA r2r8d8g8a8b-o6co5b-o6cc8.o5b-32o6c32
1350 DATA dcde-dco5b-ab-o6dcdo5f+o6dcd
1360 DATA o5go6dcdo5f+o6dcdo5b-4o6c4d4c4
1370 DATA r8o5b-8r8a8r8d8gab-g
1380 DATA r2r1
1390 DATA r2d8r8c8r8
1400 DATA d8r8c8r8.gf+gr16gf+gr16gf+g
1410 DATA d4c4o4b-4
```

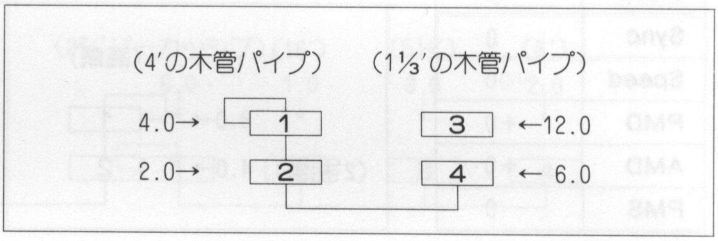
35　PORGAN　2　讃美歌判奏用パイプオルガン　oct 2〜6

35	Op1	Op2	Op3	Op4
Fb/Al	28 (3/4)			
Dt/Ml	20 (+1/4)	18 (+1/2)	12 (+0/12)	6 (+0/6)
Ol	38	30	38	0
KS/AR	146 (2/18)	18 (0/18)	94 (1/30)	89 (1/25)
DR	10	4	6	5
SR	2	0	2	0
SL/RR	71 (4/7)	8 (0/8)	25 (1/9)	9 (0/9)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・基音が8.0なので実音は7オクターブ上になる.

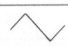
・キャリア2とモジュレータ1で矩形波を発生させ，4′の木管パイプを実現している.

・また，キャリア4とモジュレータ3も矩形波で，1⅓′の木管パイプを実現している.

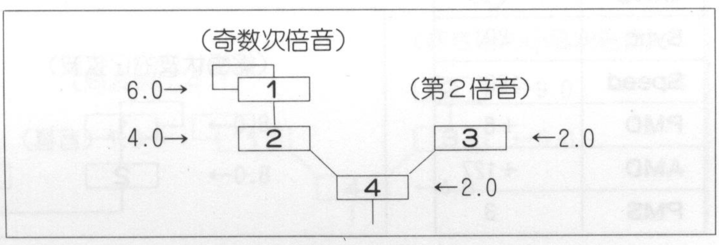



```
1000 ' 3 5  P O R G A N 2  賛美歌伴奏用パイプオルガン
1010 'prog45.bas",A
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 35,V
1090 '
1100 P$="@35"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ3，4
1240   V(2,0)=31
1250   VOICE 35,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l16o5t66
1300 DATA o4g4o5d4o4b-4.a8g8b-8a8g8
1310 DATA f+8a8d4g8d8a8d8
1320 DATA b-8aga8d8g8dga8dab-8agado5dc
1330 DATA o4b-agb-agf+agdgab-o5cder2
1340 DATA r2r8d8g8a8b-o6co5b-o6cc8.o5b-32o6c32
1350 DATA dcde-dco5b-ab-o6dcdo5f+o6dcd
1360 DATA o5go6dcdo5f+o6dcdo5b-4o6c4d4c4
1370 DATA r8o5b-8r8a8r8d8gab-g
1380 DATA r2r1
1390 DATA r2d8r8c8r8
1400 DATA d8r8c8r8.gf+gr16gf+gr16gf+g
1410 DATA d4c4o4b-4
```

36(7)　FLUTE　　上品なフルート　　oct 4〜6

36(7)	Op1	Op2	Op3	Op4
Fb/Al	43 (5/3)			
Dt/Ml	22 (+1/6)	4 (+0/4)	50 (+3/2)	2 (+0/2)
Ol	85	67	49	0
KS/AR	215 (3/23)	213 (3/21)	143 (2/15)	15 (0/15)
DR	9	7	31	31
SR	3	0	0	0
SL/RR	169 (10/9)	89 (5/9)	9 (0/9)	10 (0/10)
AMS	0	0	0	0
Wave				
Sync	8			
Speed	3000			
PMD	+45			
AMD	+127			
PMS	1			

・フルートは倍音の少ない楽器であるが，第２倍音は強く，奇数次の倍音も存在する．
・キャリア４とモジュレータ３で第２倍音を発生している．
・モジュレータ1と2は共に減衰形のエンベロープを持ちアタック時に奇数次の倍音を付加している．
・基音が3.0なので実音は2オクターブ上である．
・ビブラートとトレモロがないと物足りない感じがする．

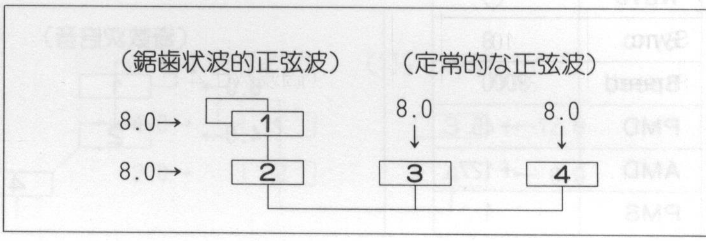



```
1000 '３６（７）　ＦＬＵＴＥ　上品なフルート
1010 'prog46.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 36,V
1090 '
1100 P$="@36"
1110 PLAY P$,P$
1120 READ A$
1130 FOR I=0 TO 1
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1)
1220 '
1230 PRINT "オペレータ３，４のみ
1240   V(2,0)=31
1250   VOICE 36,V
1260   PLAY P$(0),P$(1)
1270 END
1280 '
1290 DATA v15l8o5t132
1300 DATA ab-o6dco5b-16a16gc
1310 DATA fgab-16a16gab-o6c16o5b-16
1320 DATA ao6cccc4o5b-16a16g16f16
1330 DATA efga16g16fgab-16a16r2
1340 DATA fgb-ag16f16ec
1350 DATA fefg16f16efga16g16
1360 DATA faaaa4g16f16e16d16
1370 DATA cdef16e16defg16f16
```

37　PICCOLO　　軽くプレイしたいピッコロ　　oct 3〜6

37	Op1	Op2	Op3	Op4
Fb/Al	62 (7/6)			
Dt/Ml	8 (+0/8)	8 (+0/8)	8 (+0/8)	8 (+0/8)
Ol	40	8	30	0
KS/AR	153 (2/25)	146 (2/18)	20 (0/20)	19 (0/19)
DR	10	6	11	11
SR	15	0	0	0
SL/RR	89 (5/9)	249 (15/9)	25 (1/9)	25 (1/9)
AMS	0	0	0	15
Wave	／＼ (三角波)			
Sync	20			
Speed	180			
PMD	+6			
AMD	+127			
PMS	3			

・基音が8.0なので実音は7オクターブ上の音になる．
・キャリア2とモジュレータ1でアタック時に倍音を多く含む減衰形の鋸歯状波的正弦波を発生している．
・キャリア3と4は全く同じセッティングで，定常的な正弦波である．
・ビブラートがかけてあるが，なくても十分である．




```
1000 ' ３７　ＰＩＣＣＯＬＯ　軽くプレイしたいピッコロ
1010 'prog47.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 37,V
1090 '
1100 P$="@37"
1110 PLAY P$,P$
1120 READ A$
1130 FOR I=0 TO 1
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1)
1220 '
1230 PRINT "オペレータ３，４のみ
1240   V(2,0)=31
1250   VOICE 37,V
1260   PLAY P$(0),P$(1)
1270 END
1280 '
1290 DATA v15l8o4t132
1300 DATA ab-o6dco5b-16a16gc
1310 DATA fgab-16a16gab-o6c16o5b-16
1320 DATA ao6cccc4o5b-16a16g16f16
1330 DATA efga16g16fgab-16a16r2
1340 DATA fgb-ag16f16ec
1350 DATA fefg16f16efga16g16
1360 DATA faaaa4g16f16e16d16
1370 DATA cdef16e16defg16f16
```

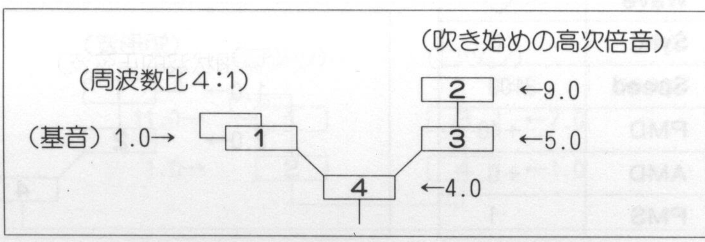
38(8)　OBOE　　ダブル・リードの代表オーボエ　oct 4〜6

38(8)	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	49 (+3/1)	57 (+3/9)	53 (+3/5)	52 (+3/4)
Ol	31	45	55	0
KS/AR	217 (3/25)	220 (3/28)	86 (1/22)	82 (1/18)
DR	11	12	0	31
SR	0	12	0	0
SL/RR	25 (1/9)	90 (5/10)	26 (1/10)	11 (0/11)
AMS	0	0	0	15
Wave				
Sync	10			
Speed	3200			
PMD	+20			
AMD	+127			
PMS	1			

・キャリア4とモジュレータ1がメイン．キャリアとモジュレータの周波数比4：1はリード楽器の基本的なセッティング

・減衰形のエンベロープを持つモジュレータ2でアタック時の高次音を付加している．

・ビブラートをかけないと少し機械的な音に聞こえる．




```
1000 ' 38 (8)  OBOE  ダブルリードの代表オーボエ
1010 'prog48.bas",A
1020 CLEAR &h600
1030 PLAY ALLOC 256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 38,V
1090 '
1100 P$="@38"
1110 PLAY P$,P$
1120 READ A$
1130 FOR I=0 TO 1
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1)
1220 '
1230 PRINT "オペレータ3, 4のみ
1240   V(1,0)=31
1250   V(2,0)=31
1260   VOICE 38,V
1270   PLAY P$(0),P$(1)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA ab-o6dco5b-16a16gc
1320 DATA fgab-16a16gab-o6c16o5b-16
1330 DATA ao6cccc4o5b-16a16g16f16
1340 DATA efga16g16fgab-16a16r2
1350 DATA fgb-ag16f16ec
1360 DATA fefg16f16efga16g16
1370 DATA faaaa4g16f16e16d16
1380 DATA cdef16e16defg16f16
```

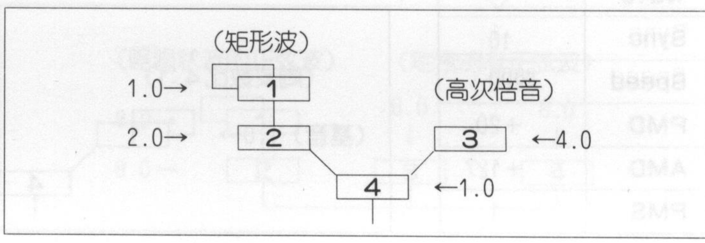
39(9)　CLARINET　ソロにむくクラリネット　oct 4〜6

39(9)	Op1	Op2	Op3	Op4
Fb/Al	59 (7/3)			
Dt/Ml	1 (+0/1)	2 (+0/2)	4 (+0/4)	1 (+0/1)
Ol	67	36	47	0
KS/AR	95 (1/31)	19 (0/19)	95 (1/31)	83 (1/19)
DR	31	11	31	31
SR	0	0	0	0
SL/RR	7 (0/0)	71 (4/7)	9 (0/9)	9 (0/9)
AMS	0	0	0	0
Wave				
Sync	6			
Speed	3100			
PMD	+40			
AMD	+0			
PMS	1			

・キャリア4とモジュレータ1・2の3段重ねで，クラリネット特有の矩形波を発生している．

・モジュレータ3は高次倍音を補う役割をしている．

・ビブラートがかけてあるが，なくても十分である．




```
1000 ' 3 9 ( 9 )   C L A R I N E T   ソロにむくクラリネット
1010 'prog49.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 39,V
1090 '
1100 P$="@39"
1110 PLAY P$,P$
1120 READ A$
1130 FOR I=0 TO 1
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1)
1220 '
1230 PRINT "オペレータ3, 4のみ
1240   V(2,0)=31
1250   VOICE 39,V
1260   PLAY P$,P$
1270   PLAY P$(0),P$(1)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA ab-o6dco5b-16a16gc
1320 DATA fgab-16a16gab-o6c16o5b-16
1330 DATA ao6cccc4o5b-16a16g16f16
1340 DATA efga16g16fgab-16a16r2
1350 DATA fgb-ag16f16ec
1360 DATA fefg16f16efga16g16
1370 DATA faaaa4g16f16e16d16
1380 DATA cdef16e16defg16f16
```

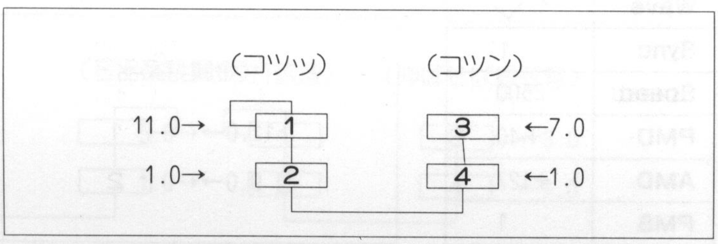
40　GROCKEN　　叩く感じがリアルなグロッケン　　oct 4〜6

40	Op1	Op2	Op2	Op4
Fb/Al	28 (3/4)			
Dt/MI	59 (+3/11)	1 (+0/1)	7 (+0/7)	49 (+3/1)
Ol	33	10	23	0
KS/AR	223 (3/31)	222 (3/30)	95 (1/31)	222 (3/30)
DR	14	9	16	7
SR	0	7	5	4
SL/RR	255 (15/15)	22 (1/6)	160 (10/0)	23 (1/7)
AMS	0	0	0	15
Wave	∧∨			
Sync	1			
Speed	2800			
PMD	+20			
AMD	+4			
PMS	2			

・2系統ともに叩くサウンドを実現しているが，モジュレータのサスティンレベルの違いにより，キャリア2は「コツッ」という短い音，キャリア4は「コツン」というやや尾を引く音である．

・グロッケンという音色名にこだわらなければoct1〜3もおもしろい．

・ビブラートがかけてあるがなくても十分である．

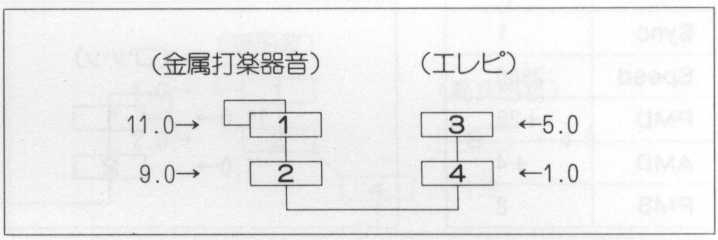



```
1000 ' ４０　ＧＲＯＣＫＥＮ　叩く感じがリアルなグロッケン
1010 'prog50.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 40,V
1090 '
1100 P$="@40q7"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ１，２のみ
1240   V(4,0)=31
1250   VOICE 40,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l8o5t132
1300 DATA o6e2d4c2d4
1310 DATA e1r4f4
1320 DATA g2f4e2f4
1330 DATA g1.r2
1340 DATA r8eo6co5eber8eo6co5ebe
1350 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1360 DATA r8go6eo5go6do5gr8go6co5go6do5g
1370 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1380 DATA o4a2.a2.
1390 DATA a2.a2.
1400 DATA o5c2.c2.
1410 DATA c2.c2.
```

41(10) VIBRPHN 上品なヴィブラフォン oct 5〜8

41(10)	Op1	Op2	Op3	Op4
Fb/Al	36 (4/4)			
Dt/MI	59 (+3/11)	25 (+1/9)	5 (+0/5)	1 (+0/1)
Ol	17	27	45	0
KS/AR	94 (1/30)	219 (3/27)	157 (2/29)	158 (2/30)
DR	24	19	12	5
SR	0	10	11	10
SL/RR	178 (11/2)	131 (8/3)	245 (15/5)	244 (15/4)
AMS	4	0	0	15
Wave	／＼			
Sync	1			
Speed	2500			
PMD	+40			
AMD	+127			
PMS	1			

・2系統が各々独立したエレピの音色でひとつのヴィブラフォンの音を実現している．
・キャリア2とモジュレータ1は中途半端な周波数比で，マレットが金属片を叩いた時に生じるサウンドを実現している．
・キャリア4とモジュレータ3はエレピのような音である．
・ビブラートがかけてあるが，少なくても十分である．



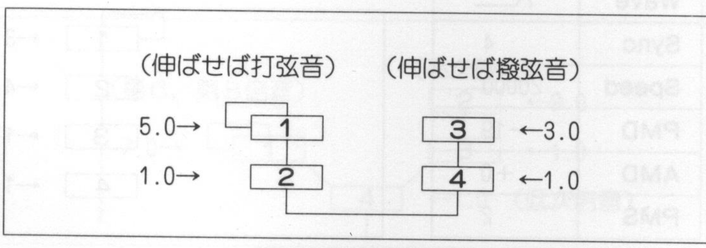

```
1000 ' 4 1  V I B R P H I N  上品なビブラフォン
1010 'prog51.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 41,V
1090 '
1100 P$="@41q7"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ1，2のみ
1240   V(4,0)=31
1250   VOICE 41,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l8o5t132
1300 DATA o6e2d4c2d4
1310 DATA e1r4f4
1320 DATA g2f4e2f4
1330 DATA g1.r2
1340 DATA r8eo6co5eber8eo6co5ebe
1350 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1360 DATA r8go6eo5go6do5gr8go6co5go6do5g
1370 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1380 DATA o4a2.a2.
1390 DATA a2.a2.
1400 DATA o5c2.c2.
1410 DATA c2.c2.
```

42　XYLOPHN　木の響きをもつシロフォン　　oct 4〜8

42	Op1	Op2	Op3	Op4
Fb/Al	60 (7/4)			
Dt/Ml	115 (+7/3)	33 (+2/1)	51 (+3/3)	49 (+3/1)
Ol	25	18	27	0
KS/AR	89 (1/25)	95 (1/31)	89 (1/25)	89 (1/25)
DR	22	13	19	13
SR	10	11	11	12
SL/RR	232 (14/8)	248 (15/8)	232 (14/8)	248 (15/8)
AMS	2	2	2	2
Wave	/\/			
Sync	1			
Speed	100			
PMD	+ 4			
AMD	+127			
PMS	1			

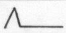
・2系統とも定常的には撥弦楽器あるいは打弦楽器の音源に用いられる周波数比であるが，エンベロープの各レートが大きく，木琴のように「コツン」としたサウンドになる．

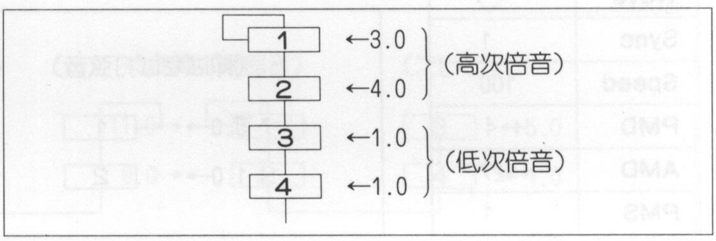
・ビブラート・ワウ・トレモロとすべてかけてあるが，なくても十分といえる．




```
1000 ' 4 2  X Y L O P H I N  木の響きを持つシロフォン
1010 'prog52.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 42,V
1090 '
1100 P$="@42q7"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ1，2のみ
1240   V(4,0)=31
1250   VOICE 42,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l8o5t132
1300 DATA o6e2d4c2d4
1310 DATA e1r4f4
1320 DATA g2f4e2f4
1330 DATA g1.r2
1340 DATA r8eo6co5eber8eo6co5ebe
1350 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1360 DATA r8go6eo5go6do5gr8go6co5go6do5g
1370 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1380 DATA o4a2.a2.
1390 DATA a2.a2.
1400 DATA o5c2.c2.
1410 DATA c2.c2.
```

43　KOTO　　日本音階で演奏するとリアルな琴　　oct 5〜7

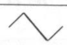
43	Op1	Op2	Op3	Op4
Fb/AL	56 (7/0)			
Dt/MI	51 (+3/3)	52 (+3/4)	49 (+3/1)	49 (+3/1)
Ol	30	30	30	0
KS/AR	218 (3/26)	221 (3/29)	220 (3/28)	223 (3/31)
DR	8	5	4	10
SR	5	4	2	3
SL/RR	39 (2/7)	20 (1/4)	54 (3/6)	21 (1/5)
AMS	0	0	0	0
Wave				
Sync	4			
Speed	20000			
PMD	−19			
AMD	+0			
PMS	2			



・弦をはじいた直後は基音がほとんどなく，高次倍音が極端に多い．
・減衰中は高次倍音が減衰していくため，低次の倍音が遅れて立上がってきたように聞こえる．
・このような弦をはじく楽器音をモジュレータの3段重ねで実現している．

```
1000 '４３　ＫＯＴＯ　日本音階で演奏するとリアルな琴
1010 'prog53.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 43,V
1090 '
1100 P$="@43q7"
1110 PLAY P$,P$,P$
1120 READ A$,B$,C$
1130 P$(0)=A$+B$+C$+B$+C$
1140 READ B$
1150 P$(1)=A$+B$
1160 '
1170 PRINT "ノーマル
1180 PLAY P$(0),P$(1)
1190 '
1200 PRINT "オペレータ３，４のみ
1210   V(2,0)=31
1220   VOICE 43,V
1230   PLAY P$(0),P$(1)
1240 END
1250 '
1260 DATA v15l16o4t40
1270 DATA e8bo5deabae4.r8o4
1280 DATA e8bo5deabo6de4.r8o4
1290 DATA r1r1o6e2g8.e32d32o5b8a8b1r4
```

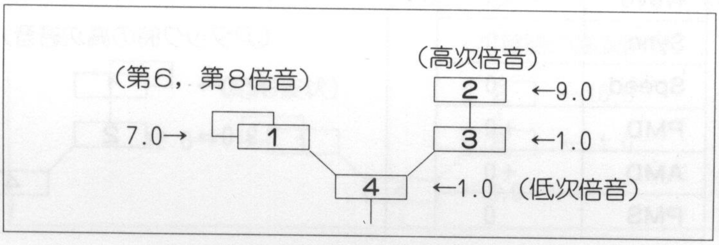
44 ZITAR	インドのシタール			oct 4〜7

44	Op1	Op2	Op3	Op4
Fb/Al	2 (0/2)			
Dt/Ml	55 (+3/7)	57 (+3/9)	33 (+2/1)	49 (+3/1)
Ol	28	69	29	5
KS/AR	159 (2/31)	159 (2/31)	159 (2/31)	159 (2/31)
DR	6	6	12	12
SR	6	6	6	6
SL/RR	1 (0/1)	1 (0/1)	1 (0/1)	5 (0/5)
AMS	0	0	0	0
Wave				
Sync	25			
Speed	3000			
PMD	+10			
AMD	+0			
PMS	1			

・キャリア4とモジュレータ3で低次の倍音を発生，さらにモジュレータ2で第9～11倍音付近を補っている．

・モジュレータ1は第6および第8倍音を発生，弦の振動音を実現している．

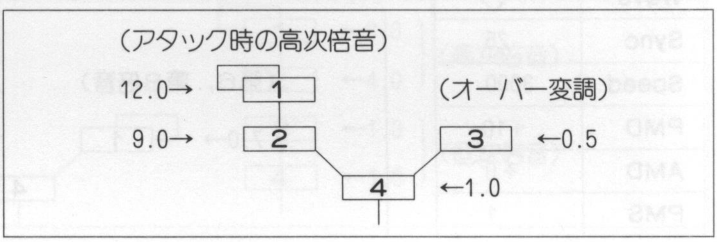
・ビブラートがかけてあるがなくても十分である．




```
1000 ' ４４  ＺＩＴＡＲ  インドのシタール
1010 'prog54.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 44,V
1090 '
1100 P$="@44"
1110 PLAY P$,P$,P$
1120 READ A$,B$,C$
1130 P$(0)=A$+B$+C$+B$+C$
1140 READ B$
1150 P$(1)=A$+B$
1160 '
1170 PRINT "ノーマル
1180 PLAY P$(0),P$(1)
1190 '
1200 PRINT "オペレータ２，３，４のみ
1210   V(1,0)=31
1220   VOICE 44,V
1230   PLAY P$(0),P$(1)
1240 END
1250 '
1260 DATA v15l16o4t40
1270 DATA e8bo5deabae4.r8o4
1280 DATA e8bo5deabo6de4.r8o4
1290 DATA r1r1o6e2g8.e32d32o5b8a8b1r4
```

45　CLAV　　ファンキーなクラビネット　　oct 4〜6

45	Op1	Op2	Op3	Op4
KS/AR	59（7/3）			
Dt/MI	60 （+3/12）	57 （+3/9）	48 （+3/0）	49 （+3/1）
Ol	33	34	7	8
KS/AR	223 （3/31）	31 （0/31）	31 （0/31）	223 （3/31）
DR	4	5	4	1
SR	4	4	4	2
SL/RR	247 （15/7）	7 （0/7）	23 （1/7）	172 （10/12）
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			



・キャリア4とモジュレータ2で偶数次のオクターブ関係の倍音を強調して打弦楽器のサウンドを創っている.

・また，キャリア4をモジュレータ3でフルに変調してオクターブ関係の倍音を発生.

・基音の12倍の周波数をもつモジュレータ1は，アタック時のみオクターブ関係にない倍音を発生している.

```
1000 '４５　ＣＬＡＶ　ファンキーなクラビネット
1010 'prog55.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 45,V
1090 '
1100 P$="@45"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ３，４のみ
1240   V(2,0)=31
1250   VOICE 45,V
1260   PLAY P$(0),P$(1),P$(2)
1270 END
1280 '
1290 DATA v15l16o5t66
1300 DATA o4g4o5d4o4b-4.a8g8b-8a8g8
1310 DATA f+8a8d4g8d8a8d8
1320 DATA b-8aga8d8g8dga8dab-8agado5dc
1330 DATA o4b-agb-agf+agdgab-o5cder2
1340 DATA r2r8d8g8a8b-o6co5b-o6cc8.o5b-32o6c32
1350 DATA dcde-dco5b-ab-o6dcdo5f+o6dcd
1360 DATA o5go6dcdo5f+o6dcdo5b-4o6c4d4c4
1370 DATA r8o5b-8r8a8r8d8gab-g
1380 DATA r2r1
1390 DATA r2d8r8c8r8
1400 DATA d8r8c8r8.gf+gr16gf+gr16gf+g
1410 DATA d4c4o4b-4
```

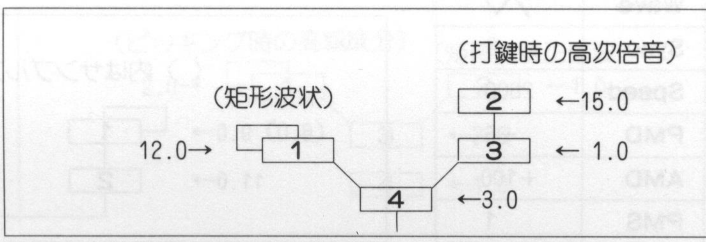
46(0)(11) HARPSIC クラシカルなチェンバロ oct 2〜6

46(0)(11)	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	12 (+0/12)	31 (+1/15)	1 (+0/1)	115 (−1/3)
Ol	32	57	30	0
KS/AR	31 (0/31)	223 (3/31)	31 (0/31)	159 (2/31)
DR	12	2	12	5
SR	4	4	4	7
SL/RR	26 (1/10)	246 (15/6)	6 (0/6)	39 (2/7)
AMS	0	0	0	0
Wave	／|			
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・チェンバロは基音より高次倍音のきわめて多い楽器である.

・キャリア4とモジュレータ3はキャリアの周波数の方が高い. これは本来のFM変調の波形でありキャリアは消滅しがちである.

・モジュレータ1はその他の倍音を補い, モジュレータ2はアタック時の倍音を発生している.




```
1000 ' 46 HARPSIC クラシカルなチェンバロ
1010 'prog56.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 46,V
1090 '
1100 P$="@46"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ3, 4のみ
1240   V(1,0)=31
1250   V(2,0)=31
1260   VOICE 46,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l16o5t66
1310 DATA o4g4o5d4o4b-4.a8g8b-8a8g8
1320 DATA f+8a8d4g8d8a8d8
1330 DATA b-8aga8d8g8dga8dab-8agado5dc
1340 DATA o4b-agb-agf+agdgab-o5cder2
1350 DATA r2r8d8g8a8b-o6co5b-o6cc8.o5b-32o6c32
1360 DATA dcde-dco5b-ab-o6dcdo5f+o6dcd
1370 DATA o5go6dcdo5f+o6dcdo5b-4o6c4d4c4
1380 DATA r8o5b-8r8a8r8d8gab-g
1390 DATA r2r1
1400 DATA r2d8r8c8r8
1410 DATA d8r8c8r8.gf+gr16gf+gr16gf+g
1420 DATA d4c4o4b-4
```

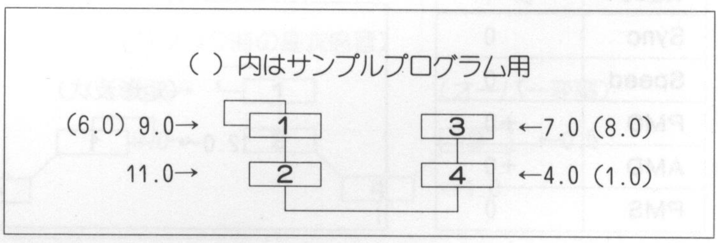
47 (12) BELL　　代表的な鐘の音　　oct 1～8

47	Op1	Op2	Op3	Op4
Fb/Al	60 (7/4)			
Dt/Ml	89 (−3/9)	43 (+2/11)	103 (−2/7)	4 (+0/4)
Ol	33	35	25	0
KS/AR	23 (0/23)	25 (0/25)	217 (3/25)	95 (1/31)
DR	20	8	20	3
SR	0	3	0	5
SL/RR	116 (7/4)	245 (15/5)	116 (7/4)	245 (15/5)
AMS	6	4	4	15
Wave				
Sync	6			
Speed	2600			
PMD	+6			
AMD	+100			
PMS	1			

・比較的音程感のある鐘の音で低い音から高い音までそれなりに使える.

・デチューンを深くかけることにより，鐘の音のうなりを実現している.

・サンプルプログラムのように音程感を失くしてもおもしろい.




```
1000 ' 4 7   B E L L   代表的な鐘の音
1010 'prog57.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 47,V
1090 P$="@47"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 1
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "音程感をなくす
1240   V(1,7)=0:V(3,7)=0:V(4,7)=0
1250   VOICE 47,V
1260 PLAY P$(0),P$(1),P$(2)
1270 END
1280 DATA v15l4o3t66
1290 DATA f+2b4.o4c+8c+2o3b2
1300 DATA o4f+ag+ef+1r2
1310 DATA f+2f+4.f+8f+2f+2
1320 DATA o4dddddl
1330 DATA d2d4.d8d2d2
1340 DATA bbbbbl
```

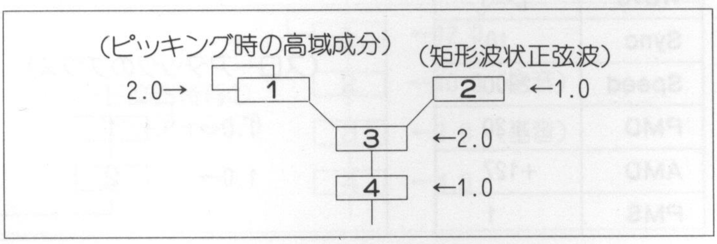
48　HARP　ピッキングのリアルなハープ　oct 5〜8

48	Op1	Op2	Op3	Op4
Fb/Al	57 (7/1)			
Dt/Ml	2 (+0/2)	1 (+0/1)	2 (+0/2)	1 (+0/1)
Ol	30	38	45	0
KS/AR	95 (1/31)	95 (1/31)	31 (0/31)	159 (2/31)
DR	12	13	6	12
SR	0	6	5	7
SL/RR	244 (15/4)	21 (1/5)	21 (1/5)	20 (1/4)
AMS	0	0	0	0
Wave	/|			
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・定常的な音色はキャリア4とモジュレータ3，2で発生する矩形波的な正弦波である．

・モジュレータ1はピッキング時の倍音構成をより矩形波的にして高域成分を強調している．

・原則的にはハープ系の音色は1：3の周波数比で発生させるので，変則的なセッティング例である．




```
1000 '４８　ＨＡＲＰ　ピッキングのリアルなハープ
1010 'prog58.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 48,V
1090 '
1100 P$="@48q7"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ３．４のみ
1240   V(1,0)=31
1250   V(2,0)=31
1260   VOICE 48,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

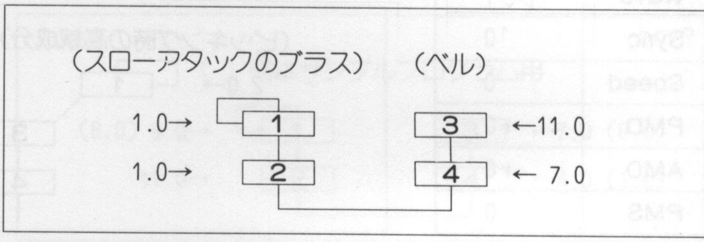
49　BELL/BRASS　使い分けできるベルとブラス　　oct 3〜6

49	Op1	Op2	Op3	Op4
Fb/Al	60（7/4）			
Dt/Ml	49 （+3/1）	1 （+0/1）	59 （+3/11）	55 （+3/7）
Ol	27	0	40	3
KS/AR	15 （0/15）	19 （0/19）	31 （0/31）	31 （0/31）
DR	11	3	10	10
SR	3	6	6	4
SL/RR	24 （1/8）	12 （0/12）	245 （15/5）	245 （15/5）
AMS	0	0	2	15
Wave				
Sync	10			
Speed	2800			
PMD	+20			
AMD	+127			
PMS	1			

・キャリア2とモジュレータ1は平凡なスローアタックのブラスだが，キャリア4とモジュレータ3のベルのエンベロープに特徴がある．

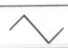
・サスティンレベルとアタック・ディケイレイトの関係で，長い音で演奏すると瞬間的に減衰してベル音は出ない．

・しかし，短い音で演奏すると音が減衰しきらないうちにサスティン状態になり，ベルの長い余韻が残る．

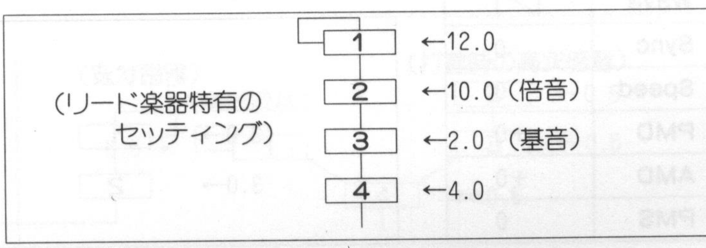



```
1000 ' ４９　ＢＲＡＳＳ　使い分けできるベルとブラス
1010 'prog59.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 49,V
1090 P$="@49"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 2
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PLAY P$(0),P$(1),P$(2)
1210 '
1220 END
1230 '
1240 DATA v15l64o3t66
1250 DATA f+2b4.o4c+8c+2o3b2
1260 DATA t132o4f+64r32.r8.r4a64r32.r8.r4
1270 DATA g+64r32.r8.r4e64r32.r8.r4f+64r1
1280 DATA f+2f+4.f+8f+2f+2
1290 DATA t132o4d64r32.r8.r4d64r32.r8.r4
1300 DATA d64r32.r8.r4d64r32.r8.r4d64
1310 DATA d2d4.d8d2d2
1320 DATA t132b64r32.r8.r4b64r32.r8.r4
1330 DATA b64r32.r8.r4b64r32.r8.r4b64
```

50 HARMONICA 小学生の吹くハーモニカ oct 2〜8

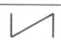
50	Op1	Op2	Op3	Op4
Fb/Al	56 (7/0)			
Dt/Ml	92 (−3/12)	10 (+0/100)	18 (+1/2)	4 (+0/4)
Ol	45	35	33	5
KS/AR	20 (0/20)	20 (0/20)	16 (0/16)	14 (0/14)
DR	5	31	31	8
SR	0	0	0	0
SL/RR	153 (9/9)	9 (0/9)	9 (0/9)	25 (1/9)
AMS	0	5	1	15
Wave				
Sync	1			
Speeb	2800			
PMD	+20			
AMD	+127			
PMS	1			

・キャリア4とモジュレータ3の周波数比2：1はリード楽器の基本的なセッティング.
・さらにモジュレータ1と2で倍音を作っていて，吹き始めをリアルに実現している.
・ビブラートが効果的にかけてあり，ビブラートがないと物足りない.



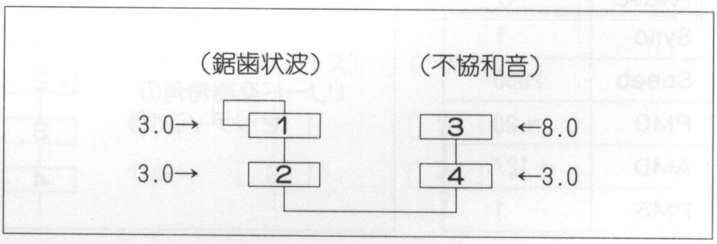

```
1000 ' 50 HARMONICA 小学生の吹くハーモニカ
1010 'prog60.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 50,V
1090 P$="@50q7"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 READ B$
1140 READ C$
1150 READ D$
1160 P$=A$+B$+B$+C$+D$
1170 '
1180 PRINT "ノーマル
1190 PLAY P$
1200 '
1210 PRINT "オペレータ3，4のみ
1220   V(2,0)=31
1230   VOICE 50,V
1240   PLAY P$
1250 END
1260 '
1270 DATA v15l16o6t60
1280 DATA c+2r16o5b(a+ag+)8g8.abo6b+
1290 DATA c+8d+8g+8e4o5g+8b4
1300 DATA r16bo6c+o5a+2.r2
```

51 STEEL DRUM スチール・ドラム oct 1〜6

51	Op1	Op2	Op3	Op4
Fb/Al	4 (0/4)			
Dt/Ml	83 (−3/3)	3 (+0/3)	56 (+3/8)	51 (+3/3)
Ol	35	10	28	0
KS/AR	86 (1/22)	153 (2/25)	84 (1/20)	153 (2/25)
DR	5	2	7	12
SR	6	6	6	6
SL/RR	101 (6/5)	22 (1/6)	101 (6/5)	102 (6/6)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・キャリア2とモジュレータ1で鋸歯状歯を発生し基音としている．そのため，実音は2オクターブ上である．

・キャリア4とモジュレータ3は不協和音的なスチールドラムのサウンドを実現している．




```
1000 ' 5 1  S T E E L  D R U M スチールドラム
1010 'prog61.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 51,V
1090 P$="@51"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0), P$(1)
1150 PRINT "[C]キーを押してください。"
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190    IF I$="" THEN 1180
1200    IF I$="c" OR I$="C" THEN 1240
1210    PLAY "c"
1220    GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H3+&HC*I
1260   VOICE 51,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o3t66
1310 DATA オペレータ1，2のみ
1320 DATA ノーマル
```

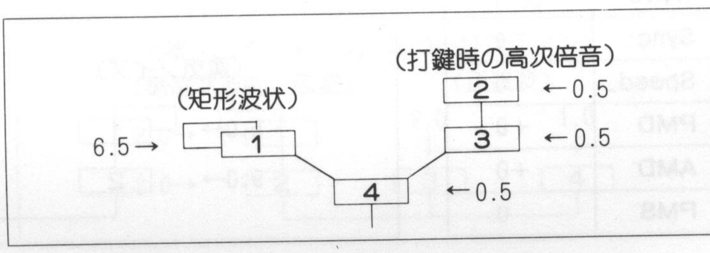
52　TIMPANI　　音程を持つ打楽器ティンパニ　　oct 1〜2

52	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	80 (−3/0)	96 (−2/0)	32 (+2/0)	0 (+0/0)
Ol	25	9	29	0
KS/AR	153 (2/25)	157 (2/29)	223 (3/31)	31 (0/31)
DR	19	13	5	5
SR	0	7	5	0
SL/RR	84 (5/4)	37 (2/5)	53 (3/5)	246 (15/6)
AMS	0	2	1	10
Wave	╲╱			
Sync	6			
Speed	300			
PMD	+10			
AMD	+30			
PMS	1			

・ティンパニの音域は狭いので注意を要する.

・キャリア4とモジュレータ2・3により，皮のノーマルな振動とベースのように音程を持った音を実現している.

・ビブラートがかけてあるが，なくても十分である.

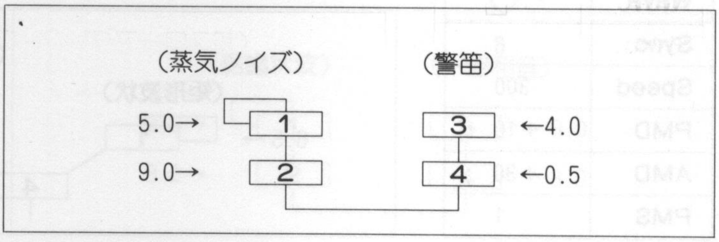



```
1000 ' ５２　ＴＩＭＰＡＮＩ　音程を持つ打楽器ティンパニ
1010 'prog62.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 52,V
1090 P$="@52"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0), P$(1)
1150 PRINT "[C]キーを押してください。"
1160 PRINT "    音色を変化させます。
1170 '
1180 I$=INKEY$
1190    IF I$="" THEN 1180
1200    IF I$="c" OR I$="C" THEN 1240
1210    PLAY "c"
1220    GOTO 1180
1230
1240 PRINT P$(I)
1250   V(0,1)=&H3*I+&HC
1260   VOICE 52,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o3t66
1310 DATA オペレータ３，４のみ
1320 DATA ノーマル
```

53　TRAIN　　なつかしい蒸気機関車　　oct 3〜5

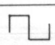
53	Op1	Op2	Op3	Op4
Fb/Al	60（7/4）			
Dt/Ml	53（+3/5）	57（+3/9）	52（+3/4）	48（+3/0）
Ol	10	21	41	0
KS/AR	77（1/13）	205（3/13）	12（0/12）	20（0/20）
DR	31	31	1	1
SR	2	2	1	1
SL/RR	8（0/8）	8（0/8）	86（5/6）	24（1/8）
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・フィードバックをかけたモジュレータ1の出力を大きくしてキャリア2を変調することによりホワイトノイズを発生させ蒸気ノイズを得ている.
・キャリア4と1′に相当するモジュレータ3で得られるオルガンサウンドのエンベロープを変えて警笛にしている.




```
1000 ' 5 3   T R A I N   なつかしい蒸気機関車
1010 'prog63.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 P$="@53"
1080 PLAY P$
1090 '
1100 READ A$
1110 READ B$
1120 READ C$
1130 '
1140 PLAY A$
1150 '
1160 FOR TEM=200 TO 250 STEP 2
1170   PLAY B$
1180 NEXT
1190 '
1200 PLAY C$
1210 '
1220 DATA 18
1230 DATA t=tem;cr8
1240 DATA o6cr8cl
```

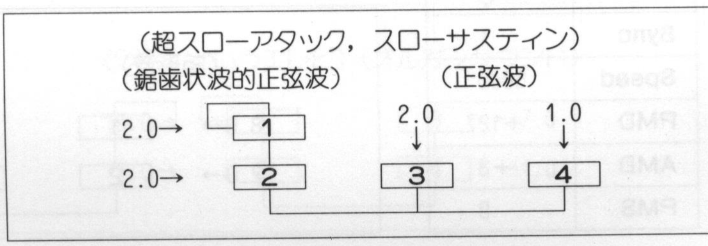
54　AMBULAN　　救急車のサイレン　　　　oct 5

54	Op1	Op2	Op3	Op4
Fb/Al	62 (7/6)			
Dt/Ml	2 (+0/2)	2 (+0/2)	2 (+0/2)	1 (+0/1)
Ol	41	23	23	0
KS/AR	15 (0/15)	15 (0/15)	15 (0/15)	9 (0/9)
DR	31	31	31	31
SR	2	2	2	2
SL/RR	5 (0/5)	5 (0/5)	5 (0/5)	5 (0/5)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	2000			
PMD	+127			
AMD	+0			
PMS	8			

・矩形波によるビブラートを利用したセッティング例.

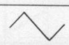
・キャリア2とモジュレータ1が鋸歯状波的正弦波，キャリア3と4も正弦波で超スローアタック，スローサスティンを持つ音である．これにより近づいて来てまた遠去かる音量効果を出している.

・また，ほぼオクターブになるように設定されたビブラートで「ピーポー」を実現している.

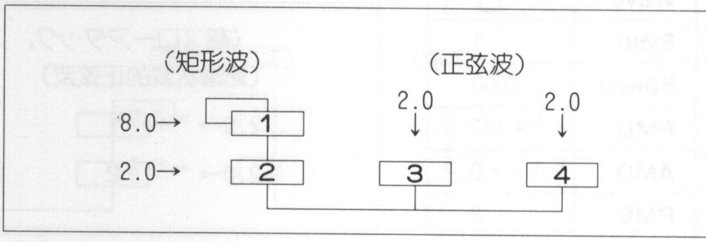



```
1000 ' 5 4  A M B U L A N   救急車のサイレン
1010 'prog64.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 VOICE INIT
1060 '
1070 READ A$
1080 PLAY A$
1090 '
1100 END
1110 '
1120 DATA @54l1o5t32a^a^a^a
```

55　TWEET　　可愛らしい小鳥のさえずり　　oct 7〜8

55	Op1	Op2	Op3	Op4
Fb/Al	62 (7/6)			
Dt/Ml	56 (+3/8)	50 (+3/2)	50 (+3/2)	50 (+3/2)
Ol	49	37	83	0
KS/AR	223 (3/31)	212 (3/20)	223 (3/31)	21 (0/21)
DR	10	6	11	11
SP	15	0	15	9
SL/RR	90 (5/10)	250 (15/10)	26 (1/10)	27 (1/11)
AMS	2	2	2	2
Wave				
Sync	1			
Speed	5000			
PMD	+127			
AMD	+3			
PMS	9			

・キャリア2とモジュレータ1で矩形波，キャリア3と4は正弦波である.
・この音に三角波でビブラートやトレモロ等をかけることにより小鳥のさえずりを実現している.
・なおキャリア3が「ピッ」というアタックを担当している.
・演奏はスタッカートで.




```
1000 ' 55  TWEET  かわいらしい小鳥のさえずり
1010 'prog65.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 VOICE INIT
1060 '
1070 READ A$
1080 PLAY A$,A$,A$
1090 '
1100 READ A$,B$,C$
1110 FOR I=0 TO 20
1120   R1=2^(RND(1)*3+1)
1130   R2=2^(RND(1)*3+1)
1140   R3=2^(RND(1)*3+1)
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @55v15l16o8t66
1210 DATA r=r1;e
1220 DATA r=r2;f
1230 DATA r=r3;f+
```

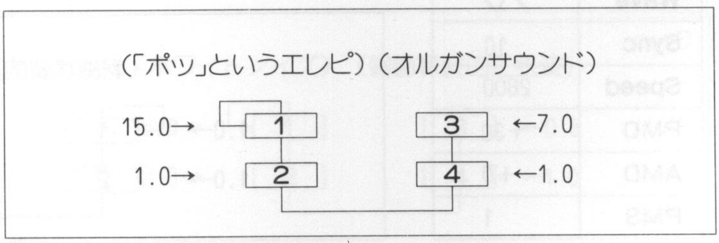
56 RAIN DROP 雨の落ちる音 oct 7〜8

56	Op1	Op2	Op3	Op4
Fb/AI	60 (7/4)			
Dt/MI	63 (+3/15)	33 (+2/1)	55 (+3/7)	49 (+3/1)
OI	33	22	30	0
KS/AR	210 (3/18)	217 (3/25)	219 (3/27)	222 (3/30)
DR	12	15	14	10
SR	10	11	11	1
SL/RR	247 (15/7)	247 (15/7)	231 (14/7)	55 (3/7)
AMS	15	15	15	15
Wave				
Sync	1			
Speed	10000			
PMD	+40			
AMD	+0			
PMS	6			

・キャリア2とモジュレータ1は，減衰の速いエレピサウンドで「ポッ」という音を出している．

・キャリア4とモジュレータ3は，ノコギリ波リンショットの頭にピークを持つビブラートで，雨音の雰囲気を出している．

・スタッカートで演奏すると雨の落ちるような感じが出る．



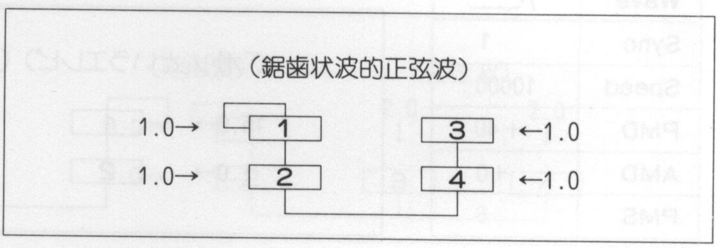

```
1000 ' 5 6   R A I N   D R O P   雨の落ちる音
1010 'prog66.bas".a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 VOICE INIT
1060 '
1070 READ A$
1080 PLAY A$,A$,A$
1090 '
1100 READ A$,B$,C$
1110 FOR I=0 TO 20
1120   R1=2^(RND(1)*3+1)
1130   R2=2^(RND(1)*3+1)
1140   R3=2^(RND(1)*3+1)
1150   PLAY A$,B$,C$
1160 NEXT
1170 '
1180 END
1190 '
1200 DATA @56v15o6t66
1210 DATA r=r1;e
1220 DATA r=r2;f
1230 DATA r=r3;f+
```

57　HORN　　柔らかい響きが心地良いホルン　　oct 3〜6

57	Op1	Op2	Op3	Op4
Fb/Al	52 (6/4)			
Dt/Ml	1 (+0/1)	1 (+0/1)	1 (+0/1)	1 (+0/1)
Ol	30	0	31	8
KS/AR	16 (0/16)	16 (0/16)	16 (0/16)	16 (0/16)
DR	1	2	1	2
SR	0	0	0	0
SL/RR	24 (1/8)	24 (1/8)	24 (1/8)	24 (1/8)
AMS	0	0	0	0

Wave	
Sync	10
Speed	2800
PMD	+30
AMD	+0
PMS	1

・ほとんど同じ音色を2系列で発生させている.
・キャリアとモジュレータの周波数比が1：1なので鋸歯状波的正弦波である.
・アタックの遅い独持のエンベロープにより吹き始めは鋭く，しだいに丸みを帯びた鋸歯状波に移る音色はホルンそのものである.




```
1000 '５７　ＨＯＲＮ　柔らかい響きが心地よいホルン
1010 'prog67.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 57,V
1090 '
1100 P$="@57q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ P$
1140 FOR I=0 TO 3
1150   READ A$:P$=P$+A$
1160 NEXT
1170 '
1180 PRINT "ノーマル
1190 PLAY P$
1200 PRINT "オペレータ１，２のみ
1210   V(4,0)=31
1220   VOICE 57,V
1230   PLAY P$
1240 END
1250 '
1260 DATA v15l8o3t80
1270 DATA bo4def+gaf+dd2
1280 DATA o3bo4de4.f+gabo5d
1290 DATA c+o4af+2o5c+o4a
1300 DATA f+4.bgeco3ao4d+3bb2.r2
```

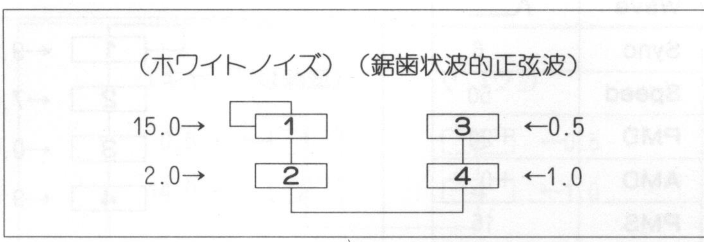
58 SNARE DRUM デモでおなじみスネアドラム　oct 5

58	Op1	Op2	Op3	Op4
Fb/Al	60 (7/4)			
Dt/Ml	15 (+0/15)	2 (+0/2)	0 (+0/0)	1 (+0/1)
Ol	12	15	19	0
KS/AR	223 (3/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	31	15	24	19
SR	0	17	0	16
SL/RR	8 (0/8)	44 (2/12)	184 (11/8)	44 (2/12)
Wave	0	0	0	0
Sync				
Speed	0			
PMD	0			
AMD	+0			
AMD	+0			
PMS	0			

・キャリア2とモジュレータ1はホワイトノイズを発生，スネアドラムの中心となる音を実現している．

・キャリア4とモジュレータ3は超低音の鋸歯状波的正弦波で皮の振動音を発生している．

・音域が狭いので注意．



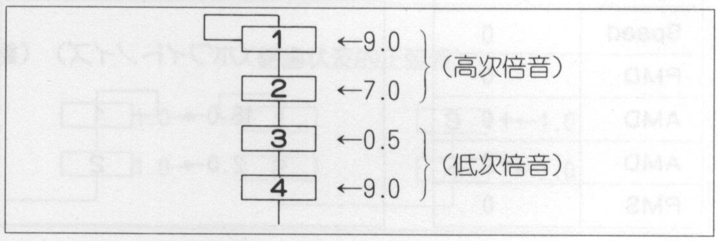

```
1000 '５８　ＳＮＡＲＥ　ＤＲＵＭ　デモでおなじみスネアドラム
1010 'prog68.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 58,V
1090 P$="@58"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "　　　　音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&HC*I+&H3
1260   VOICE 58,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o5t66
1310 DATA オペレータ１，２のみ
1320 DATA ノーマル
```

59　COW　BELL　叩く感覚でプレイするカウベル　oct 4〜5

59	Op1	Op2	Op3	Op4
Fb/AL	56 (7/0)			
Dt/MI	9 (+0/19)	7 (+0/7)	0 (+0/0)	1 (+0/1)
OI	7	0	37	0
KS/AR	223 (3/31)	223 (3/31)	23 (0/23)	23 (0/23)
DR	19	19	19	31
SR	13	13	15	15
SL/RR	92 (5/12)	92 (5/12)	92 (5/12)	11 (0/11)
AMS	0	0	0	0
Wave	Λ			
Sync	6			
Speed	50			
PMD	+20			
AMD	+0			
PMS	15			

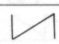
・モジュレータ1とモジュレータ2で金属的ノイズ音を発生させウベルの音色の中心としている.

・モジュレータ3とキャリア4で「コツ」というアタックの音を発生させている.



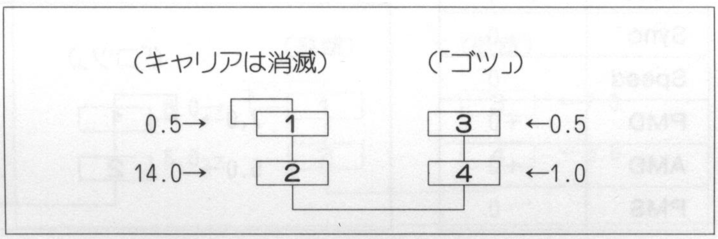

```
1000 '59　COW　BELL　叩く感覚でプレイするカウベル
1010 'prog69.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 59,V
1090 P$="@59"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H3*I+&HC
1260   VOICE 59,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o5t66
1310 DATA オペレータ3，4のみ
1320 DATA ノーマル
```

60　PERC　1　　ひょうきんなパーカッシヴトーン　　oct 1〜5

60	Op1	Op2	Op3	Op4
Fb/Al	52（6/4）			
Dt/MI	0 （+0/0）	14 （+0/14）	180 （−3/0）	17 （+1/1）
Ol	13	20	29	0
KS/AR	31 （0/31）	31 （0/31）	31 （0/31）	95 （1/31）
DR	1	13	22	20
SR	0	15	7	20
SL/RR	37 （2/5）	250 （15/10）	104 （6/8）	248 （15/8）
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・キャリア２とモジュレータ１は本来のＦＭ変調であり，キャリアの周波数の両側にサイドバンドが現われキャリア自体は消滅しがちである．このため音程感が失なわれる．

・キャリア４とモジュレータ３は非常に減衰を速くして「ゴツ」というアタック音を出している．




```
1000 ' 6 0   0 E R C 1   ひょうきんなパーカッシブトーン
1010 'prog70.bas",a
1020 CLEAR &h600
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 60,V
1090 P$="@60"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "    音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H3*I+&HC
1260   VOICE 60,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o5t66
1310 DATA オペレータ３，４のみ
1320 DATA ノーマル
```

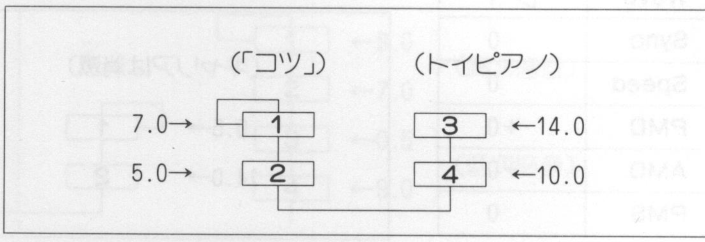
61　PERC　2　可愛らしいパーカッシヴトーン　　oct 1～8

61	Op1	Op2	Op3	Op4
Fb/Al	52 (6/4)			
Dt/Ml	7 (＋0/7)	5 (＋0/5)	94 (−3/14)	58 (＋3/10)
Ol	23	10	22	0
KS/AR	30 (0/30)	31 (0/31)	223 (3/31)	223 (3/31)
DR	19	14	19	15
SR	30	28	13	10
SL/RR	216 (13/8)	27 (1/11)	76 (4/12)	25 (1/9)
AMS	0	0	0	0
Wave	/|/|			
Sync	0			
Speed	0			
PMD	＋0			
AMD	＋0			
PMS	0			

・2つの系列はオクターブが異なるだけで，基本的にはキャリアとモジュレータの周波数比が5：7のため音程感の失なわれたサウンドである．

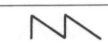
・減衰形のエンベロープを持つので，高域ではトイピアノのように聞こえる．

・キャリア2とモジュレータ1が「コツ」というアタック音，キャリア4とモジュレータ3が減衰の速いトイピアノサウンドである．



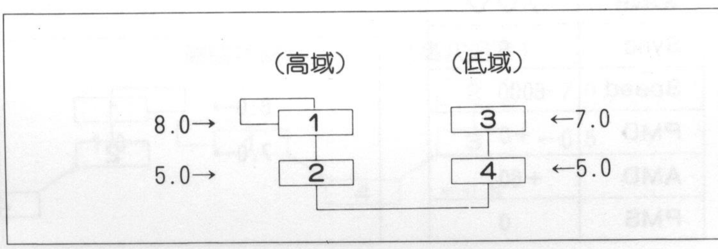

```
1000 ' 6 1  P E R C 2  かわいらしいパーカッシブトーン
1010 'prog71.bas",a
1020 CLEAR &H600
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 61,V
1090 P$="@61"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H3*I+&HC
1260   VOICE 61,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o5t66
1310 DATA オペレータ3，4のみ
1320 DATA ノーマル
```

62　PERC　3　小憎らしいパーカッシブトーン　oct3〜6

62	Op1	Op2	Op3	Op4
Fd/Al	60 (7/4)			
Dt/Ml	40 (+2/8)	53 (+3/5)	55 (+3/7)	5 (+0/5)
Ol	25	10	26	20
KS/AR	29 (0/29)	29 (0/29)	31 (0/31)	31 (0/31)
DR	12	12	12	12
SR	0	0	0	0
SL/RR	246 (15/6)	246 (15/6)	246 (15/6)	246 (15/6)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・PERC 2とほぼ同様だが，モジュレータ1とキャリア2が高域を受け持つ点が異なる．

・4つのオペレータのエンベロープがほぼ一致しているため，高域と低域で同時に鳴っている感じがする．

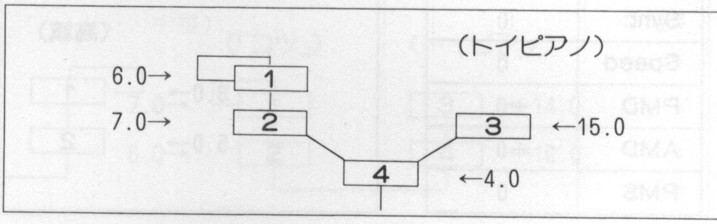



```
1000 ' 62 PERC3 小憎らしいパーカッシブトーン
1010 'prog72.bas".a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 62,V
1090 P$="@62"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "      音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H3*I+&HC
1260   VOICE 62,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o5t66
1310 DATA オペレータ3，4のみ
1320 DATA ノーマル
```

63　MUSIC BOX　食器を並べてたたいたようなチャイム　oct3〜6

63	Op1	Op2	Op3	Op4
Fd/Al	59 (7/3)			
Dt/Ml	54 (+3/6)	7 (+0/7)	95 (−3/15)	4 (+0/4)
Ol	5	35	17	0
KS/AR	220 (3/28)	31 (0/37)	220 (3/28)	31 (0/31)
DR	17	19	19	13
SR	11	0	11	11
SL/RR	39 (2/7)	119 (7/7)	247 (15/7)	246 (15/6)
AMS	10	5	0	15
Wave	/\/\/			
Sync	8			
Speed	6000			
PMD	+0			
AMD	+60			
PMS	0			

・モジュレータ３とキャリア４によるトイピアノのような音とモジュレータ１とモジュレータ２による音程感のない音が組み合わさっている．




```
1000 '６３　ＭＵＳＩＣ　ＢＯＸ　食器を並べて叩いたようなチャイム
1010 'prog73.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 63,V
1090 P$="@63"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H2*I+&HD
1260   VOICE 63,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l16o5t66
1310 DATA オペレータ３．４のみ
1320 DATA ノーマル
```

64　CELLO　オンザロックの似合うチェロ　oct4〜6

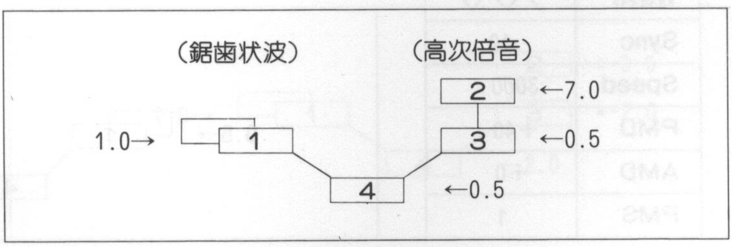
64	Op1	Op2	Op3	Op4
Fd/Al	58 (7/2)			
Dt/Ml	17 (+1/1)	55 (+3/7)	32 (+2/0)	16 (+1/0)
Ol	28	30	40	0
KS/AR	83 (1/19)	83 (1/19)	81 (1/17)	75 (1/11)
DR	10	16	16	31
SR	0	0	0	0
SL/RR	23 (1/7)	58 (3/10)	24 (1/8)	8 (0/8)
AMS	0	0	0	0
Wave	/\/\/			
Sync	8			
Speed	3200			
PMD	+36			
AMD	+0			
PMS	1			

・モジュレータ1と3がキャリア4を変調して，鋸歯状波を発生．

・モジュレータのアタックは早く，ディケイは遅い．

・モジュレータ2は弾き始めの高次倍音を受け持っている．

・基音が½なので実音は1オクターブ下となる．



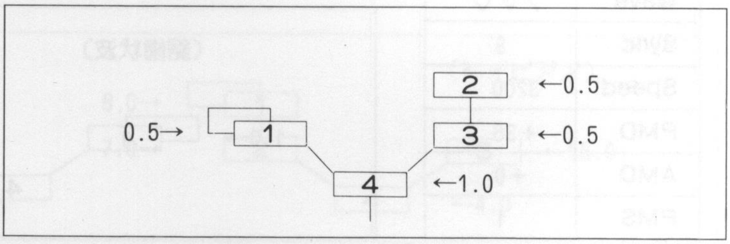

```
1000 ' ６４　ＣＥＬＬＯ　オンザロックの似合うチェロ
1010 'prog74.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 64,V
1090 P$="@64"
1100 PLAY P$,P$,P$
1110 '
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 2
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル"
1210 PLAY P$(0),P$(1),P$(2)
1220 PRINT "オペレータ２，３，４のみ"
1230   V(1,0)=31
1240   VOICE 64,V
1250   PLAY P$(0),P$(1),P$(2)
1260 END
1270 '
1280 DATA v15l8o5t132
1290 DATA r8ab-o6dco5b-16a16gcfgab-16a16g2
1300 DATA r2r8g16a16b-gar4b-gg16a16b-g
1310 DATA a4o6c4o5b-4.ao6d4c4o5gg16a16b-gr2
1320 DATA r1efg16f16e2
1330 DATA ro4ab-o5dc2r8o4ab-o5dc2
1340 DATA c4f4e4.fb-4a4e2
1350 DATA c1.c2.
1360 DATA f4d4e4.cf4d4e4.c
1370 DATA f4o6c4c4.ff4c4c2
```

65　LOW BRASS　低音の効いた金管サウンド　oct2〜4

65	Op1	Op2	Op3	Op4
Fd/AI	58 (7/2)			
Dt/MI	16 (+1/10)	80 (−3/0)	48 (+3/0)	1 (+0/1)
OI	31	43	31	0
KS/AR	14 (0/14)	15 (0/15)	15 (0/15)	17 (0/17)
DR	4	4	8	0
SR	0	0	0	12
SL/RR	39 (2/7)	55 (3/7)	55 (3/7)	56 (3/8)
AMS	0	0	0	0
Wave				
Sync	10			
Speed	3000			
PMD	+40			
AMD	+0			
PMS	1			

・モジュレータ1が，キャリア4を変調して金管特有の割れるようなサウンドを実現している．

・他のオペレータとパターンの異なるエンベロープを持つキャリア4により，吹き初めの金属的な成分と，残る余韻を分離している．

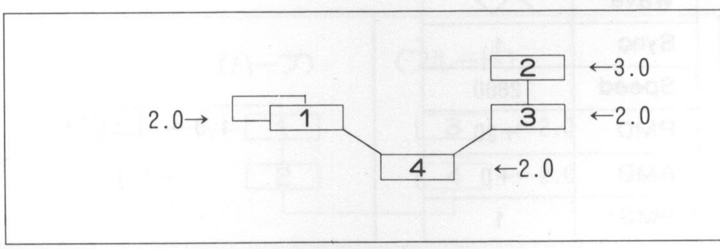



```
1000 '６５　ＬＯＷ　ＢＲＡＳＳ　低音の効いた金管サウンド
1010 'prog75.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 65,V
1090 '
1100 P$="@65"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 1
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ２，３，４のみ
1250 V(1,0)=31
1260 VOICE 65,V
1270 PLAY P$(0),P$(1),P$(2)
1280 END
1290 DATA v15l4o3t66
1300 DATA f+2b4.o4c+8c+2o3b2
1310 DATA o4f+ag+ef+1r2
1320 DATA f+2f+4.f+8f+2f+2
1330 DATA o4ddddd1
1340 DATA d2d4.d8d2d2
1350 DATA bbbbb1
```

66	WW ENSNBL	木管楽器のアンサンブル	oct4〜6

66	Op1	Op2	Op3	Op4
Fd/Al	58 (7/2)			
Dt/Ml	2 (+0/2)	35 (+2/3)	98 (−2/2)	18 (+1/2)
Ol	40	40	42	0
KS/AR	141 (2/13)	79 (1/15)	21 (0/21)	82 (1/18)
DR	6	8	7	4
SR	2	0	0	0
SL/RR	24 (1/8)	24 (1/8)	40 (2/8)	40 (2/8)
AMS	0	0	0	0
Wave	/\/\/			
Sync	12			
Speed	3200			
PMD	+34			
AMD	+0			
PMS	1			

・モジュレータ1とモジュレータ3がキャリア4を変調し，鋸歯状波を発生．
・基音が2.0なので実音は1オクターブ上となる．
・少しずつ異なるエンベロープを持つオペレータが，アンサルブルのような感じを出している．



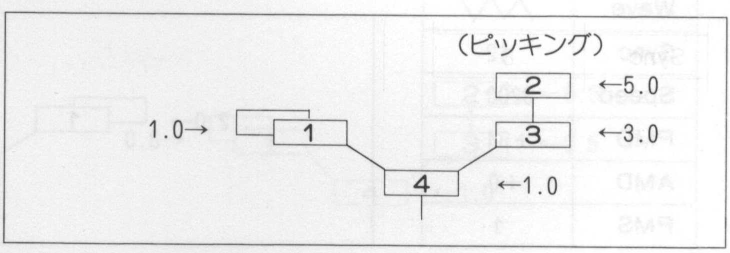

```
1000 ' ＷＷ　ＥＮＳＮＢＬ　木管楽器のアンサンブル
1010 'prog76.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 66,V
1090 '
1100 P$="@66"
1110 PLAY P$,P$
1120 READ A$
1130 FOR I=0 TO 1
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1)
1220 '
1230 PRINT "オペレータ２，３，４のみ
1240   V(1,0)=31
1250   VOICE 66,V
1260   PLAY P$(0),P$(1)
1270 END
1280 '
1290 DATA v15l8o5t132
1300 DATA ab-o6dco5b-16a16gc
1310 DATA fgab-16a16gab-o6c16o5b-16
1320 DATA ao6cccc4o5b-16a16g16f16
1330 DATA efga16g16fgab-16a16r2
1340 DATA fgb-ag16f16ec
1350 DATA fefg16f16efga16g16
1360 DATA faaaa4g16f16e16d16
1370 DATA cdef16e16defg16f16
```

67　AC GUITAR　アコーステックギターのサウンド　oct 4〜6

67	Op1	Op2	Op3	Op4
Fd/Al	58 (7/2)			
Dt/Ml	81 (−3/1)	37 (+2/5)	3 (+0/3)	1 (+0/1)
Ol	33	51	43	0
KS/AR	218 (3/26)	155 (2/17)	220 (3/28)	31 (0/31)
DR	9	3	3	5
SR	7	0	0	0
SL/RR	244 (15/4)	244 (15/4)	244 (15/4)	247 (15/7)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	2600			
PMD	+20			
AMD	+0			
PMS	1			

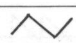
・キャリア4とモジュレータ3で，撥弦系楽器特有の第2，第4倍音系の欠落したサウンドを実現している．

・モジュレータ2は，さらにピッキング時の高めの音を作り出している．




```
1000 ' 6 7  A C  G U I T A R  アコースティックギターのサウンド
1010 'prog77.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 67,V
1090 '
1100 P$="@67q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ1，3，4のみ
1250   V(2,0)=31
1260   VOICE 67,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

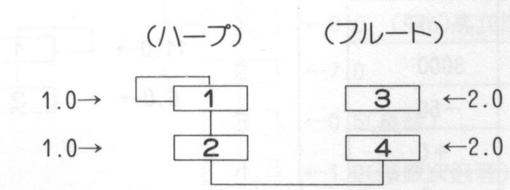
68　FLUTE/HARP　フルートとハープの見事なユニゾン　oct4～6

68	Op1	Op2	Op3	Op4
Fd/Al	60 (7/4)			
Dt/Ml	1 (+0/1)	1 (+0/1)	34 (+2/2)	2 (+0/2)
Ol	36	0	35	0
KS/AR	220 (3/28)	31 (0/31)	83 (1/19)	15 (0/15)
DR	5	9	11	31
SR	9	0	0	0
SL/RR	244 (15/4)	246 (15/6)	25 (1/9)	11 (0/11)
AMS	0	0	10	15
Wave				
Sync	8			
Speed	2600			
PMD	+0			
AMD	+60			
PMS	0			

・モジュレータ3とキャリア4がフルート系の音を，モジュレータ1とキャリア2がハープ系の音を担当している．

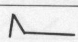
・モジュレータ1は，ハープのピッキング時の音域成分を作り出す．

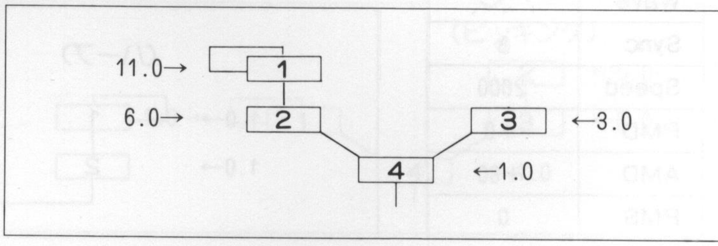
・アルゴリズム4は，このように2つの音色を作り出すときに適している．




```
1000 ' 68  FLUTE/HARP  フルートとハープの見事なユニゾン
1010 'prog78.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256.256.256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 68,V
1090 '
1100 P$="@68q7"
1110 PLAY P$.P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1).P$(2)
1220 '
1230 PRINT "オペレータ1, 2のみ
1240   V(3,0)=31
1250   V(4,0)=31
1260   VOICE 68,V
1270   PLAY P$(0).P$(1).P$(2)
1280 END
1290 '
1300 DATA v1518o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

69　FUNC PERC　輪ゴムをはじくような打弦楽器　oct 4〜6

69	Op1	Op2	Op3	Op4
Fd/Al	59 (7/3)			
Dt/Ml	27 (+1/11)	54 (+3/6)	51 (+3/3)	1 (+0/1)
Ol	48	55	12	3
KS/AR	208 (3/16)	31 (0/31)	18 (0/18)	223 (3/31)
DR	8	12	12	31
SR	0	0	0	2
SL/RR	246 (15/6)	246 (15/6)	246 (15/6)	7 (0/7)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	3000			
PMD	−50			
AMD	+0			
PMS	1			



・モジュレータ3が輪ゴムをはじくような音を立ち上がりに付加している。

・モジュレータ1とモジュレータ2が，たたくような音を立ち上がりに少しだけ付加している。

・キャリア4の出力レベル3により，モジュレータ3の効果を強くしている。

```
1000 '　６９　ＦＵＮＣ　ＰＥＲＣ　輪ゴムをはじくような打弦楽器
1010 'prog79.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 69,V
1090 P$="@69"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "　　音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H4*I+&HB
1260   VOICE 69,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ３をカット
1320 DATA ノーマル
```

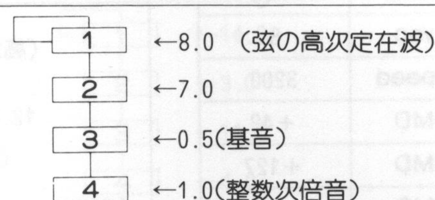
70　FUNC BASS　ファンキーなベース　oct 4〜6

70	Op1	Op2	Op3	Op4
Fd/Al	32 (4/0)			
Dt/Ml	56 (+3/8)	55 (+3/7)	48 (+3/0)	49 (+3/1)
Ol	16	60	12	0
KS/AR	223 (3/31)	223 (3/31)	144 (2/16)	159 (2/31)
DR	7	6	9	6
SR	7	6	6	8
SL/RR	41 (2/9)	25 (1/9)	25 (1/9)	249 (15/9)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・基音はモジュレータ3．よって実音は1オクターブ下となる．

・整数次倍音がモジュレータ3とキャリア4によって発生し，弦の振動する感じを出している．

・モジュレータ1と2は，高次倍音を発生し，弦のはじく感じを出している．



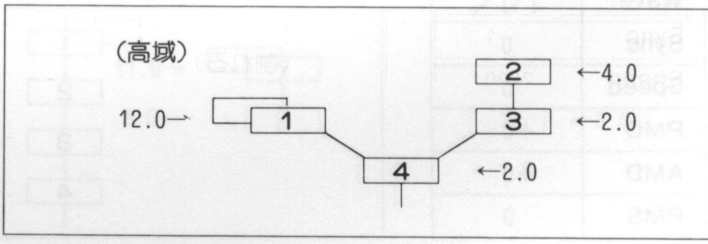

```
1000 ' 7 0  F U N C  B A S S  ファンキーなベース
1010 'prog80.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 70,V
1090 '
1100 P$="@70q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ3，4のみ
1250   V(2,0)=31
1260   VOICE 70,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

71　SYN LEAD　メロディライン向けのシンセリード　oct3〜6

71	Op1	Op2	Op3	Op4
Fd/Al	58 (7/2)			
Dt/Ml	12 (+0/12)	8 (+0/8)	2 (+0/2)	2 (+0/2)
Ol	40	50	30	0
KS/AR	14 (0/14)	22 (0/22)	22 (0/22)	29 (0/29)
DR	8	8	8	31
SR	0	0	0	0
SL/RR	246 (15/6)	54 (3/6)	54 (3/6)	8 (0/8)
AMS	0	0	1	0
Wave				
Sync	8			
Speed	3200			
PMD	+42			
AMD	+127			
PMS	1			

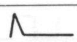
・キャリア4とモジュレータ2の周波数比1：2はリード楽器のセッティング.

・モジュレータ1が高域でのきらびやかな感じを出している.



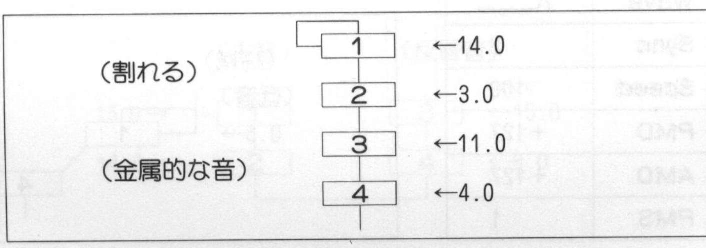

```
1000 ' 7 1  S Y N  L E A D  メロディライン向けのシンセリード
1010 'prog81.bas",A
1020 CLEAR &H200
1030 PLAY ALLOC 256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 71,V
1090 '
1100 P$="@71"
1110 PLAY P$,P$
1120 READ A$
1130 FOR I=0 TO 1
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1)
1220 '
1230 PRINT "オペレータ3, 4のみ
1240   V(1,0)=31
1250   V(2,0)=31
1260   VOICE 71,V
1270   PLAY P$(0),P$(1)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA ab-o6dco5b-16a16gc
1320 DATA fgab-16a16gab-o6c16o5b-16
1330 DATA ao6cccc4o5b-16a16g16f16
1340 DATA efga16g16fgab-16a16r2
1350 DATA fgb-ag16f16ec
1360 DATA fefg16f16efga16g16
1370 DATA faaaa4g16f16e16d16
1380 DATA cdef16e16defg16f16
```

72　METAL CLES　割れるような金属音　　　oct 2〜4

72	Op1	Op2	Op3	Op4
Fd/Al	56 (7/0)			
Dt/Ml	110 (−2/14)	3 (+0/3)	59 (+3/11)	4 (+0/4)
Ol	21	13	37	0
KS/AR	31 (0/31)	7 (0/7)	31 (0/31)	9 (0/9)
DR	31	31	7	7
SR	0	0	0	0
SL/RR	3 (0/3)	4 (0/4)	165 (10/5)	85 (5/5)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	2000			
PMD	−40			
AMD	+0			
PMS	1			

・モジュレータ3とキャリア4だけでは山状のエンベロープを持つただの金属音である。

・モジュレータ1とモジュレータ2が，ピークの遅れた山状のエンベロープを持ち，半端な周波数比で割れるような金属音を付加している。

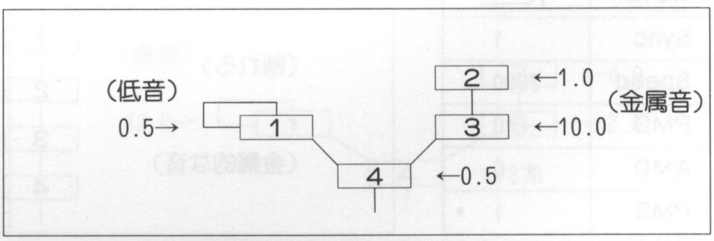



```
1000 ' ７２　ＭＥＴＡＬ　ＣＲＥＳ　割れるような金属音
1010 'prog82.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 72,V
1090 P$="@72"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H3*I+&HC
1260   VOICE 72,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l8o5t66
1310 DATA オペレータ３，４のみ
1320 DATA ノーマル
```

73 STAIN 飛び跳ねるような金属音 oct 2〜4

73	Op1	Op2	Op3	Op4
Fd/AI	58 (7/2)			
Dt/MI	48 (+3/0)	33 (+2/1)	90 (−3/10)	0 (+0/0)
OI	25	17	29	0
KS/AR	5 (0/5)	13 (0/13)	25 (0/25)	31 (0/31)
DR	5	31	5	31
SR	3	0	11	0
SL/RR	51 (3/3)	85 (5/5)	119 (7/7)	4 (0/4)
AMS	5	5	5	0
Wave				
Sync	8			
Speed	100			
PMD	+127			
AMD	+127			
PMS	1			

・モジュレータ1が遅れてわきあがる低音を受け持つ.
・モジュレータ2とモジュレータ3が，立ち上がりの鋭い金属的な音を受け持つ.
・ワンショット形のLFOがこのサウンドの特徴を出している.




```
1000 '７３　ＳＴＡＩＮ　飛び跳ねるような金属音
1010 'prog83.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 73,V
1090 P$="@73"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "    音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H4*I+&HB
1260   VOICE 73,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ１，４のみ
1320 DATA ノーマル
```

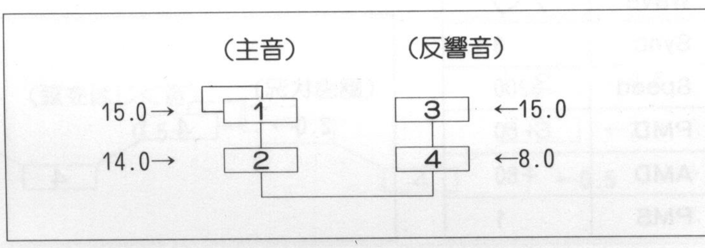
74　CUBIN GRASS 反響音を伴った金属音　oct 4〜6

74	Op1	Op2	Op3	Op4
Fd/Al	60 (7/4)			
Dt/MI	63 (+3/15)	46 (+2/14)	63 (+3/15)	8 (+0/8)
Ol	17	0	35	8
KS/AR	31 (0/31)	31 (0/31)	19 (0/19)	19 (0/19)
DR	19	19	11	11
SR	11	11	19	19
SL/RR	165 (10/5)	165 (10/5)	167 (10/7)	167 (10/7)
AMS	0	0	10	15
Wave	⊓⊔			
Sync	6			
Speed	6300			
PMD	+0			
AMD	+80			
PMS	0			

・モジュレータ１とキャリア２が“コッ”という音を，モジュレータ３とキャリア４が金属的な反響音を伴う音を実現している．

・みようによっては異ったサウンドの結合にもなる．

・距形波による浅いＬＦＯが不定感を出している．




```
1000 ' 7 4   C U B I N   G R A S S   反響音を伴った金属音
1010 'prog84.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 74,V
1090 P$="@74"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "  音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H8*I+&H7
1260   VOICE 74,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ1, 2のみ
1320 DATA ノーマル
```

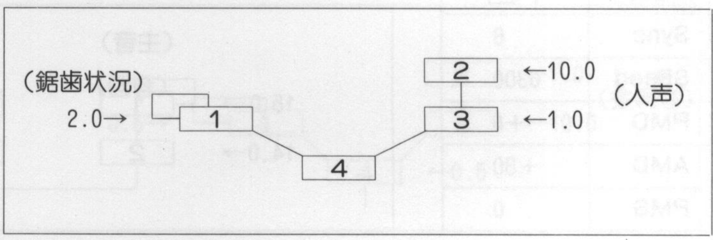
75　HUMAN　　ヴァイオリンの真似をする人の声　oct 5〜7

75	Op1	Op2	Op3	Op4
Fd/Al	58 (7/2)			
Dt/MI	18 (+1/2)	106 (−2/10)	1 (+0/1)	2 (+0/2)
Ol	45	23	49	0
KS/AR	145 (2/17)	223 (3/31)	143 (2/15)	11 (0/11)
DR	31	21	31	31
SR	0	0	0	0
SL/RR	10 (0/10)	74 (4/10)	26 (1/10)	9 (0/9)
AMS	1	2	4	0
Wave				
Sync	3			
Speed	3200			
PMD	+50			
AMD	+80			
PMS	1			

・モジュレータ1が鋸歯状波を発生し，弦楽器のような音を作り出している．

・モジュレータ1をマスクすれば，女性のソプラノとみることもできる．

・ゆるやかなビブラートが，人声のふるえる感じをよく出している．




```
1000 ' 7 5   H U M A N   バ イ オ リ ン の 真 似 を す る 人 の 声
1010 'prog85.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 75,V
1090 P$="@75"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H4*I+&HB
1260   VOICE 75,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ2，3，4のみ
1320 DATA ノーマル
```

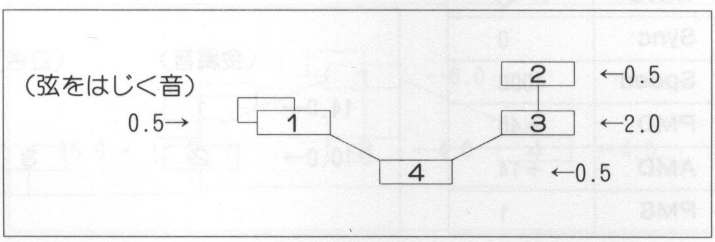
76　WOOD BASS　低音の渋いウッドベース　oct 2〜4

76	Op1	Op2	Op3	Op4
Fb/Al	42 (5/2)			
Dt/Ml	48 (+3/0)	32 (+2/0)	34 (+2/2)	0 (+0/0)
Ol	43	41	49	0
KS/AR	223 (3/31)	154 (2/26)	218 (3/26)	28 (0/28)
DR	7	5	5	5
SR	11	0	0	0
SL/RR	247 (15/7)	119 (7/7)	167 (10/7)	119 (7/7)
AMS	0	10	15	10
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			

・リード楽器のようなセッティングである.

・基音が0.5であるので，実音は1オクターブ下となる.

・モジュレータ1が，ピッキングの感じをよく出している.



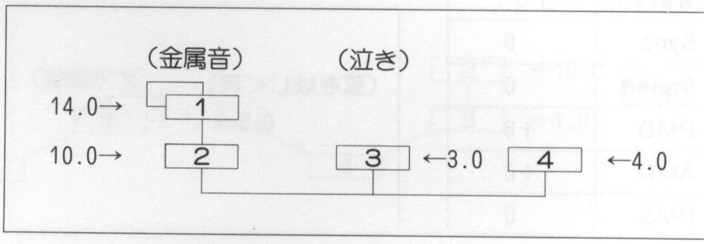

```
1000 ' 7 6  W O O D  B A S S  低音の渋いウッドベース
1010 'prog86.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 76,V
1090 '
1100 P$="@76q7"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 3
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ1，4のみ
1250   V(3,0)=31
1260   VOICE 76,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

77　CHIMES　　揺れるようなチャイム　　oct 4〜6

77	Op1	Op2	Op3	Op4
Fd/Al	62 (7/6)			
Dt/Ml	30 (+1/14)	58 (+3/10)	38 (+2/6)	4 (+0/4)
Ol	35	21	7	0
KS/AR	31 (0/31)	15 (0/15)	21 (0/21)	31 (0/31)
DR	11	11	11	11
SR	11	11	11	11
SL/RR	245 (15/5)	244 (15/4)	245 (15/5)	246 (15/6)
AMS	2	2	0	0
Wave				
Sync	0			
Speed	4000			
PMD	+40			
AMD	+14			
PMS	1			

・モジュレータ1が立ち上がり時の金属的な音を出し，キャリア3が泣くような音を実現している．

・LFOによるビブラートが，チャイムに揺れるような感じを加えている．




```
1000 ' 7 7  C H I M E S  揺れるようなチャイム
1010 'prog87.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 77,V
1090 P$="@77"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。"
1160 PRINT "    音色を変化させます。"
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H5*I+&HA
1260   VOICE 77,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ2, 4のみ
1320 DATA ノーマル
```

78 SPACY　機械的な口笛　oct 4〜6

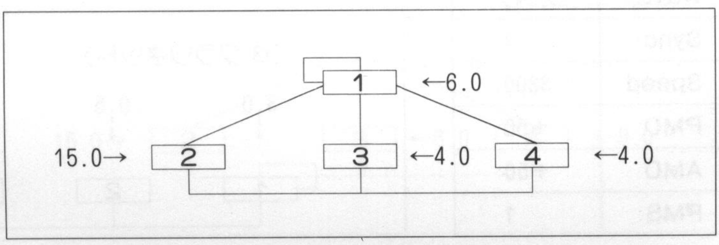
78	Op1	Op2	Op3	Op4
Fd/Al	5 (0/5)			
Dt/Ml	54 (+3/6)	31 (+1/15)	52 (+3/4)	52 (+3/4)
Ol	72	37	37	0
KS/AR	154 (2/26)	213 (3/21)	213 (3/21)	11 (0/11)
DR	5	3	3	31
SR	0	0	0	0
SL/RR	18 (1/2)	35 (2/3)	21 (1/5)	8 (0/8)
AMS	3	5	1	2
Wave				
Sync	3			
Speed	3200			
PMD	+50			
AMD	−10			
PMS	1			

・キャリア3とキャリア4はそれだけでは人声（ソプラノ）的な音を出す.

・キャリア2は高域で鳴るような感じを付加.

・モジュレータ1がほかのすべてのオペレータを変調し，無機的な感じを表現している.

・LFOが効果的にかけてあり，揺れる感じをうまく表現している.




```
1000 ' 78 SPACY 機械的な口笛
1010 'prog88.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 78,V
1090 P$="@78"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "    音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H1*I+&HE
1260   VOICE 78,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ2，3，4のみ
1320 DATA ノーマル
```

79　OBOE/B.CLAR　オーボエとバスクラリネットのユニゾン　oct 4〜6

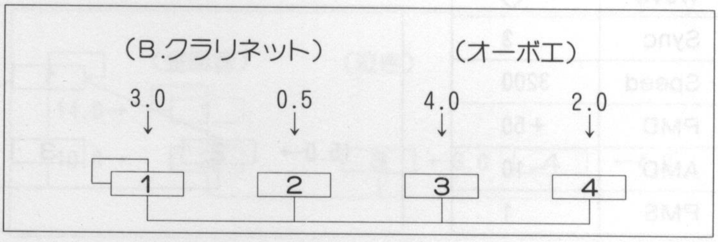
79	Op1	Op2	Op3	Op4
Fd/Al	7 (0/7)			
Dt/Ml	51 (+3/3)	48 (+3/0)	36 (+2/4)	18 (+1/2)
Ol	7	0	7	0
KS/AR	205 (3/13)	139 (2/11)	199 (3/6)	139 (2/11)
DR	31	31	31	31
SR	0	0	0	0
SL/RR	7 (0/7)	8 (0/8)	9 (0/9)	11 (0/11)
AMS	0	3	3	0
wave	／＼／			
Sync	7			
Speed	3300			
PMD	+50			
AMD	+50			
PMS	1			

・キャリア3とキャリア4がオーボエ，キャリア1とキャリア2がバスクラリネットを担当している．

・各オペレータが並列関係にあり，倍音構成で音色を特徴づける例．

・キャリア3とキャリア4は2：1の周波数比でリードらしさを出している．

・オペレータ1のフィードバックが0という点にすごい特徴がある．




```
1000 ' ７９　ＯＢＯＥ／Ｂ．ＣＬＡＲ　オーボエとバスクラリネットのユニゾン
1010 'prog89.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 79,V
1090 '
1100 P$="@79q7"
1110 PLAY P$,P$,P$
1120 READ A$
1130 FOR I=0 TO 2
1140   P$(I)=A$
1150   FOR J=0 TO 3
1160     READ B$:P$(I)=P$(I)+B$
1170   NEXT
1180 NEXT
1190 '
1200 PRINT "ノーマル
1210 PLAY P$(0),P$(1),P$(2)
1220 '
1230 PRINT "オペレータ３，４のみ
1240   V(1,0)=31
1250   V(2,0)=31
1260   VOICE 79,V
1270   PLAY P$(0),P$(1),P$(2)
1280 END
1290 '
1300 DATA v15l8o5t132
1310 DATA o6e2d4c2d4
1320 DATA e1r4f4
1330 DATA g2f4e2f4
1340 DATA g1.r2
1350 DATA r8eo6co5eber8eo6co5ebe
1360 DATA r8eo6co5eo6co5er8eo6co5eo6do5g
1370 DATA r8go6eo5go6do5gr8go6co5go6do5g
1380 DATA r8go6eo5go6eo5gr8go6eo5go6eo5g
1390 DATA o4a2.a2.
1400 DATA a2.a2.
1410 DATA o5c2.c2.
1420 DATA c2.c2.
```

80　OLD STRING　哀愁をそそるバイオリンサウンド　oct 3〜5

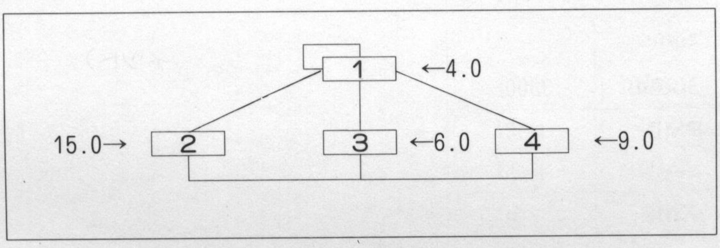
80	Op1	Op2	Op3	Op4
Fd/Al	45 (5/5)			
Dt/Ml	36 (+2/4)	63 (+3/15)	54 (+3/6)	57 (+3/9)
Ol	77	12	12	12
KS/AR	198 (3/6)	134 (2/6)	134 (2/6)	134 (2/6)
DR	31	31	31	31
SR	0	0	0	0
SL/RR	8 (0/8)	10 (0/10)	10 (0/10)	10 (0/10)
AMS	2	4	4	4
Wave	∿			
Sync	1			
Speed	3000			
PMD	+60			
AMD	+0			
PMS	2			

・キャリア２，３，４の半端な周波数比による非整数次倍音がこの音色の特徴.

・スローなアタックが，ゆっくりとしたテンポの演奏に適する.

・モジュレータ１は，高域のひっかけるような感じを出しているが弱い.

・ＬＦＯが効果的にかけてあり，情感をかもし出している.

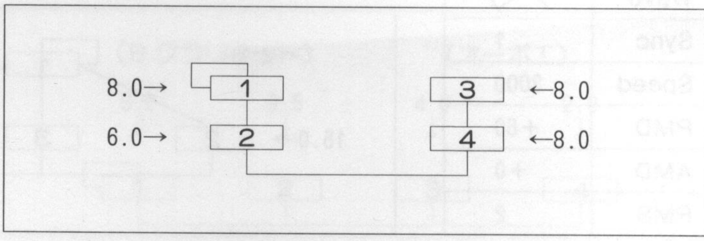



```
1000 ' 8 0  O L D  S T R I N G  哀愁をそそるバイオリン
1010 'prog90.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 80,V
1090 '
1100 P$="@80"
1110 PLAY P$,P$,P$
1120 '
1130 READ A$
1140 FOR I=0 TO 2
1150   P$(I)=A$
1160   FOR J=0 TO 1
1170     READ B$:P$(I)=P$(I)+B$
1180   NEXT
1190 NEXT
1200 '
1210 PRINT "ノーマル
1220 PLAY P$(0),P$(1),P$(2)
1230 '
1240 PRINT "オペレータ２，３，４のみ
1250 V(1,0)=31
1260 VOICE 80,V
1270 PLAY P$(0),P$(1),P$(2)
1280 END
1290 DATA v15l4o3t66
1300 DATA f+2b4.o4c+8c+2o3b2
1310 DATA o4f+ag+ef+1r2
1320 DATA f+2f+4.f+8f+2f+2
1330 DATA o4ddddd1
1340 DATA d2d4.d8d2d2
1350 DATA bbbbb1
```

81　STEEL'S CRY　鉄板をしならせるような音　oct 3〜6

81	Op1	Op2	Op3	Op4
Fd/Al	60 (7/4)			
Dt/Ml	56 (+3/8)	38 (+2/6)	56 (+3/8)	56 (+3/8)
Ol	17	7	27	0
KS/AR	31 (0/31)	25 (0/25)	19 (0/19)	25 (0/25)
DR	19	19	11	11
SR	11	11	19	19
SL/RR	165 (10/5)	165 (10/5)	167 (10/7)	167 (10/7)
AMS	3	5	13	15
Wave				
Sync	6			
Speed	6300			
PMD	+30			
AMD	+80			
PMS	10			

・モジュレータ1とキャリア2が立ち上がりの鉄をたたくような音を，モジュレータ3とモジュレータ4がしなるような反響音を実現.
・スピードの高いLFOが振動のようすをうまく出している.




```
1000 ' ８１　鉄板をしならせるような金属音
1010 'prog91.bas",a
1020 CLEAR &H100
1030 PLAY ALLOC 256
1040 DEFINT A-Z
1050 OPTION BASE 0
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 81,V
1090 P$="@81"
1100 PLAY P$
1110 '
1120 READ A$
1130 PLAY A$
1140 READ P$(0),P$(1)
1150 PRINT "[C]キーを押してください。
1160 PRINT "     音色を変化させます。
1170 '
1180 I$=INKEY$
1190   IF I$="" THEN 1180
1200   IF I$="c" OR I$="C" THEN 1240
1210   PLAY "c"
1220   GOTO 1180
1230 '
1240 PRINT P$(I)
1250   V(0,1)=&H2*I+&HD
1260   VOICE 81,V
1270   I=1-I
1280   GOTO 1180
1290 '
1300 DATA v15l2o5t66
1310 DATA オペレータ３，４のみ
1320 DATA ノーマル
```

第4章

音創りの応用

4.1 内蔵音色のエディット

PCには、69種の音色がプリセットされています。それぞれ完成度の高いものですが、自分なりに修正して、お気にいりのサウンドに作り換えるのも楽しいものです。この章では、プリセットボイスをエディットするこつを解説し、その実例を挙げてみます。

4．1．1 音色のエディット法

○ **アルゴリズム（Al）**

もっとも簡単な方法で、アルゴリズムナンバーを変更するだけです。この方法は、特にどうしようということはわからないがとにかく少し音を変えてみたい、というときに有効です。

アルゴリズムは8種ありますから、瞬時に7つの異なる音が得られます。このうち気にいったものを選び、必要があればさらに修正を加えます。

理論的でなく、偶然に頼る方法ですので、まちがいなく望みどおりの音色が得られるという保証はありませんが、労せずして意表をついた音色を得られることもあります。エディトというより、プリセットボイスを8倍に増やす手段、と考えるべきかもしれません。

○ **アウトプットレベル（Ol）**

FM音源の場合、音色の大部分は、モジュレータのOlで決まってしまいますから、これを少し変えるだけでもかなり変わった音色が得られます。

一般的に、固い音、明るい音にしたいときには、モジュレータのOlを上げ（大きい数値にする）、柔らかい、おちついた音にしたいときには下げます（小さな数値にする）。フィードバックのかかっているオペレータ、周波数値の大きいオペレータなどは音色に与える影響が大きく、音が固すぎるというようなときには、それらのOlを少し下げるだけで、おちついた音色にすることができます。

○ **フィードバック（Fb）**

アウトプットレベルの場合と同様に、固めの音にしたい場合は大きく、柔らかめの音にしたい場合は小さくします。

○ **エンベロープジェネレータ（EG）**

EGには、AR、DR、SR、RR、SLと、計5つのパラメータがありますが、もっとも重要なのは、音の立ち上がりを決めるARと減衰を決めるRRです。

たとえばピアノ系の音は、ARはきわめて速くRRはゆっくり、と比較的単純ですが、ブラス系の音は、ブラス特有の「ホワン」という感じのアタックを得るために、キャリアのARは速く、モジュレータのARは少し遅くする必要があります。また、ストリングス系の音は、逆に、モジュレータのARを速くキャリアのARはゆっくりにして、立ち上がりがおちついた感じになるようにしますが、速く細かいフレーズを弾くときなどはキャリアのARもある程度速くないとフレーズがきちんと再現できません。

このように、楽器にはそれぞれ特有のEGがありますので、しっかり把握して、最低限それだけは満たすようにしなければなりません。どんな音色であってもEGがピアノの特徴を持っていなければ、けしてピアノの音には聞こえないのです。

そのうえで、用途や好みにあわせ、AR、RRを少しづつ変えて、求める音に近づけていきます。

○ Detune、LFO

音に厚みや広がりを持たせたい場合、音色そのものよりも、どうやって複数の音源が鳴っている感じを表わすか、という問題になります。つまり、複数の楽器によって生じる周期的音量変化（トレモロ）、周期的音色変化（ワウワウ）、周期的ピッチ変化（ビブラート）を加えてやればよいわけです。

トレモロには、キャリアどうしにDetuneをかける方法、キャリアにAMDをかける方法があります。ワウワウには、キャリアとモジュレータの間にDetuneをかける方法、モジュレータにAMDをかける方法があります。また、ビブラートには、キャリアにPMDをかける方法があります。

いずれも、効果を大きくしたければ、各パラメータの絶対値を大きくすればよいのですが、大きくしすぎると逆に不自然な感じになったり、ピッチが聞きとれなくなったりしますので注意してください。

効果音の類の場合、アルゴリズムの変更と同様に偶然に頼る方法ではありますが、LFOのWaveを変えてみるだけでも、面白い音が得られることがあります。

○ マルチプル（Ml）

音色とはそもそも倍音成分と密接な関係があり、モジュレータの周波数がその倍音成分を決定しています。キャリアの周波数をf_C、モジュレータの周波数をf_Mとすると、基音f_Cとともに、$f_C \pm f_M$、$f_C \pm 2f_M$、$f_C \pm 3f_M$、…という周波数の倍音が発生することになるのです。

f_Cとf_Mは、

$$f_C = (\text{F-Num} + \text{Detune}_C) * \text{Ml}_C$$

$$f_M = (\text{F-Num} + \text{Detune}_M) * \text{Ml}_M$$

で定まりますから、モジュレータのMlを変更することによって、音色は容易に変わります。

もっとも、MIをどうすればどういう音になる、というようなはっきりした効果があるわけではありませんが、しかし、おおむね次のようなことが言えます。

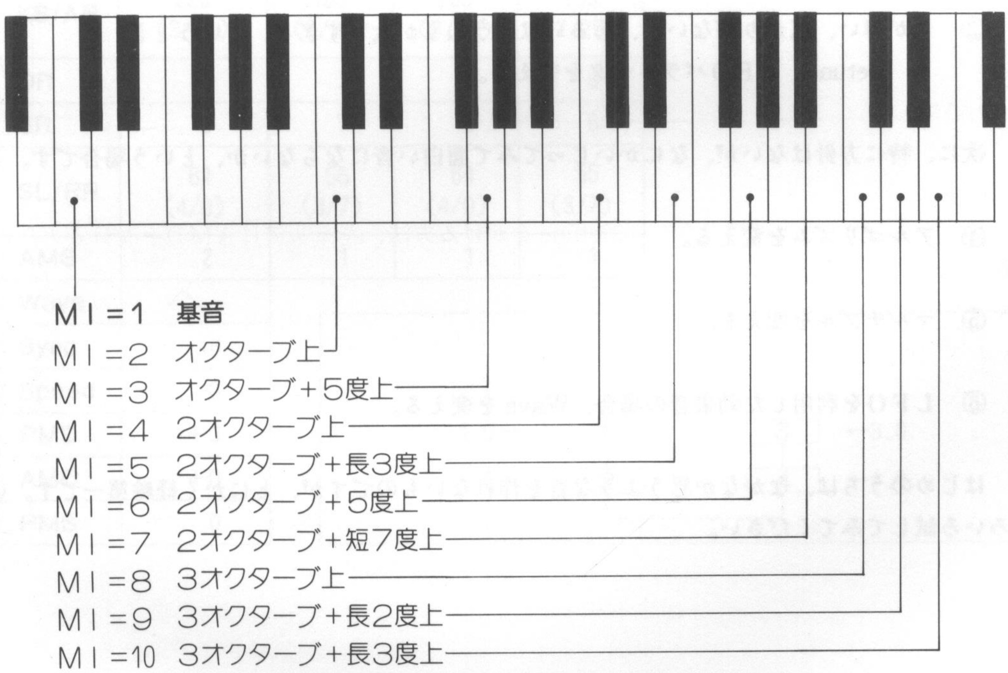
・f_C：f_M = 1：1とするのが最も一般的である。(特殊な効果を狙う場合を除く)

・f_C：f_M = 1：2とすると、矩形波に近い波形になり、木管的な音になる。

・f_Cとf_Mの比が1：8などとかなり大きくなったり、2：7のように非整数の場合、音が分散してしまうため、普通はそのような設定はしない。

キャリアのMIは、ピッチが変わるだけですが、複数のキャリアをもつアルゴリズムの場合、各キャリアのMIを異なる値に設定することにより、オルガンのカップラー的な効果が得られます。小さめの整数比であれば、異なるMIでもうまくまじりあい、新しい音色を得ることができるわけです。

参考までに、MIと音の高さの関係を示しておきます。



○ まとめ

簡単にまとめてみましょう。まず、音をどのように変えたいか、という目標がはっきりしている場合です。

① 音色が固すぎる、われている、品がない、きつすぎる、おちつけたい…、あるいは、音色が甘すぎる、丸い、地味でハリがない、柔らかすぎる…、というとき。

モジュレータのOl、またはFbを変える。

② 音の立ち上がりが速すぎる、遅すぎる、あるいは、音の切れが悪い、スパリと切れすぎる…、というとき。

ＡＲ、ＲＲを変える。

③ 音が薄い、広がりがない…、あるいは、うねりが大きすぎる、というとき。

Detune、ＬＦＯパラメータを変える。

次に、特に方針はないが、なにかいじってみて面白い音にならないか、という場合です。

④ アルゴリズムを変える。

⑤ マルチプルを変える。

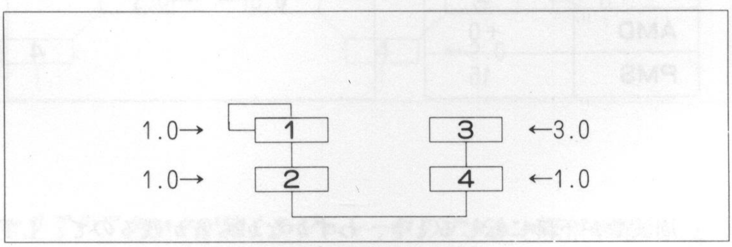
⑥ ＬＦＯを利用した効果音の場合、Waveを変える。

はじめのうちは、なかなか思うような音を作れないものですが、とにかく経験第一です。いろいろ試してみてください。

4．1．2　内蔵音色のエディト例

Edit PIANO	アコースティック・ピアノ	oct 2〜8

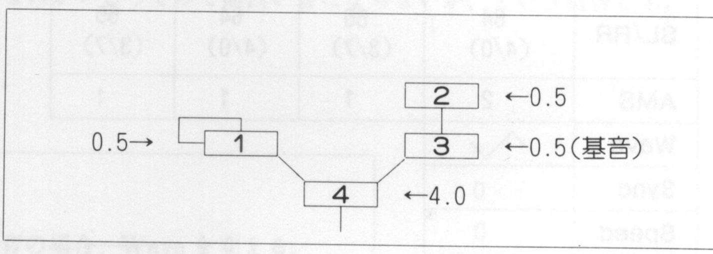
E13	Op1	Op2	Op3	Op4
Fb/Al	60（7/4）			
Dt/Ml	81 （−1/1）	17 （+1/1）	35 （+2/3）	97 （−2/1）
Ol	32	10	40	0
KS/AR	223 （3/31）	153 （2/25）	223 （3/31）	153 （2/25）
DR	0	7	0	8
SR	8	7	8	6
SL/RR	64 （4/0）	55 （3/7）	64 （4/0）	55 （3/7）
AMS	2	1	1	1
Wave	/\/			
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			



わりに整ったピアノですが、高域ではややキラキラしすぎていますので、Op 3の出力を下げて、おちつけてみました。

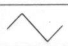
Edit　GRANPRI　　レーシングカーのエキゾースト　oct 4

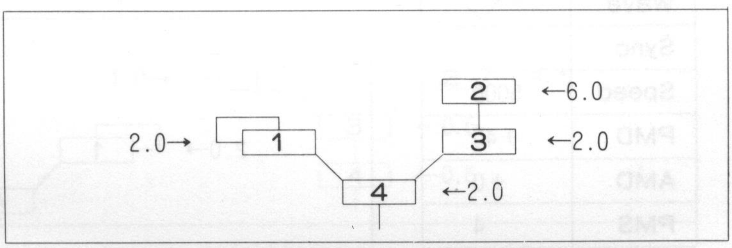
E17	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	0 (+0/0)	48 (+3/0)	48 (+3/0)	52 (+3/4)
Ol	14	20	6	0
KS/AR	76 (1/12)	80 (1/16)	157 (2/29)	137 (2/9)
DR	5	4	6	4
SR	1	1	5	31
SL/RR	85 (5/5)	37 (2/5)	37 (2/5)	168 (10/8)
AMS	0	0	0	0
Wave	／|			
Sync	1			
Speed	80			
PMD	+100			
AMD	+0			
PMS	15			



周波数が下降に転じるとき、わずかながら音が残るので、ＬＦＯを少しいじってみました。

Edit　BRASS　1　ホルン系のブラスサウンド　oct 2〜4

E21	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	2 (+0/2)	38 (+2/6)	2 (+0/2)	18 (+1/2)
Ol	25	32	42	0
KS/AR	140 (2/12)	79 (1/15)	21 (0/21)	95 (1/31)
DR	6	8	7	4
SR	2	0	0	0
SL/RR	24 (1/8)	24 (1/8)	40 (2/8)	40 (2/8)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	4000			
PMD	+10			
AMD	+0			
PMS	1			

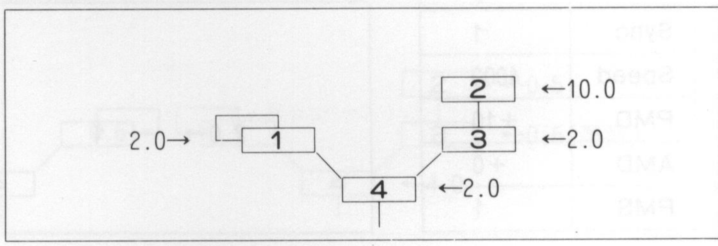


ブラス系の音の特徴は、キャリアのアタックが速くモジュレータのアタックはやや遅い、というエンベロープを持つことです。

キャリアであるOp4のARを最大にし、モジュレータであるOp1のARを少し下げて、ほどよい音色変化が得られるようにしました。

Edit　STRING　2　ストリングスキーボードのサウンド　oct 4〜6

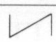
25	Op1	Op2	Op3	Op4
Fb/Al	58（7/2）			
Dt/Ml	18 （+1/2）	26 （+1/10）	2 （+0/2）	18 （+1/2）
Ol	36	24	51	0
KS/AR	78 （1/14）	83 （1/19）	92 （1/28）	77 （1/13）
DR	7	9	7	4
SR	0	0	0	0
SL/RR	19 （1/3）	50 （3/2）	19 （1/3）	7 （0/7）
AMS	0	0	0	0
Wave				
Sync	1			
Speed	5000			
PMD	+40			
AMD	+0			
PMS	4			

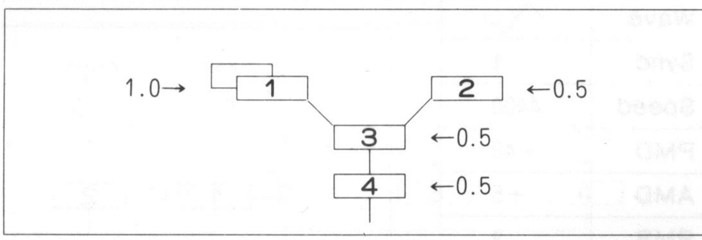


フィードバック付きのオペレータは音全体に対して大きな影響があります。

Op 1の出力を少し下げ、ARを遅くするだけで、ガサツな感じがなくなり、自然な立ち上がりになりました。

Edit　EBASS　2　代表的なエレクトリックベース　oct 3

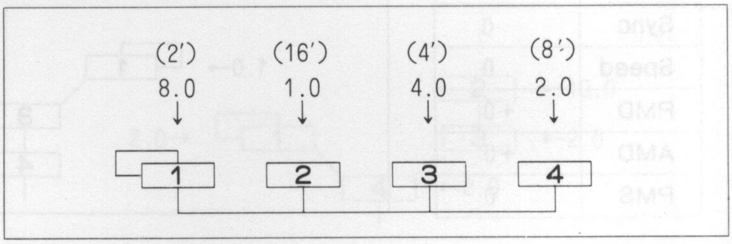
E31	Op1	Op2	Op3	Op4
Fb/A1	41 (5/1)			
Dt/MI	49 (+3/1)	48 (+3/0)	48 (+3/0)	48 (+3/0)
Ol	23	63	23	0
KS/AR	158 (2/30)	216 (3/24)	220 (3/28)	220 (3/28)
DR	14	10	4	5
SR	8	8	8	8
SL/RR	182 (11/6)	182 (11/6)	182 (11/6)	182 (11/6)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	+0			
AMD	+0			
PMS	0			



Alを変えてみました。それだけでもかなり変わった感じになりますが、さらにOp 1の周波数を上げて、明るくシンセベースのような音にしています。それだけではややギラついた感じになるので、Fbを下げて音をおちつけてあります。

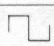
Edit EORGAN 2 16′, 8′, 4′に2′のパーカッシブ oct 4〜8

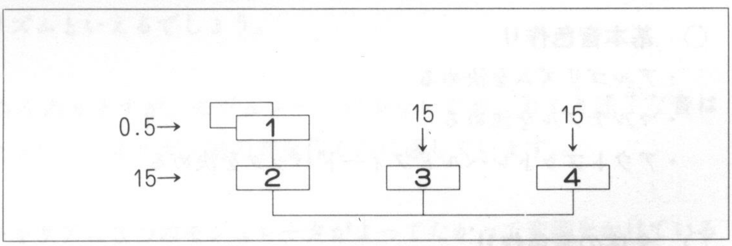
E33	Op1	Op2	Op3	Op4
Fb/AI	31 (3/7)			
Dt/MI	120 (−3/8)	49 (+3/1)	116 (−3/4)	50 (+3/2)
OI	17	10	10	0
KS/AR	28 (0/28)	159 (2/31)	31 (0/31)	31 (0/31)
DR	18	15	15	15
SR	0	0	0	0
SL/RR	255 (15/15)	15 (0/15)	15 (0/15)	15 (0/15)
AMS	0	0	0	0
Wave	〜			
Sync	1			
Speed	4400			
PMD	+40			
AMD	+5			
PMS	3			



Op 1、Op 3 の周波数を変えてみました。もともとは、基音と同時に、オクターブ下、5度上、オクターブ5度上の倍音がでていましたが、これを、オクターブ下、オクターブ上、2オクターブ上に変えました。

Edit AMBULAN	救急車のサイレン	oct 4

E54	Op1	Op2	Op3	Op4
Fb/Al	60（7/6）			
Dt/Ml	2 （+0/0）	15 （+0/15）	15 （+0/15）	15 （+0/15）
Ol	48	29	27	0
KS/AR	4 (0/4)	4 (0/4)	4 (0/4)	0 (0/0)
DR	0	0	0	0
SR	2	2	2	2
SL/RR	5 (0/5)	5 (0/5)	5 (0/5)	5 (0/5)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	900			
PMD	+44			
AMD	+0			
PMS	8			



ビブラートが大きすぎるので、PMDを下げて３度ほどにおさえました。
さらに、Mlを操作して疑似ＬＦＯを作り、音量変化を出しています。

4．2　イニットボイスからの音創り

さて、それではいよいよ自分で最初から音を作ってみることにしましょう。やはり、自分の手でオリジナルボイスを作り上げるのが、腕の見せどころ、個性を発揮する場、といえると思います。

ひとつ大事なことは、常に目的意識を持っていること、つまり、自分が作りたい音のイメージをはっきりさせておくことです。そして、その音を作り出すにはどのようなエンベロープが必要で、どのアルゴリズムが適しているか、というようにひとつづつ段階を追って進めていく姿勢が重要です。ＦＭ音源はやみくもにパラメータをいじっているうちに良い音が見つかるようなものではありません。きちんと作業を進めないと、音がグシャグシャになったり、まったく出なくなったりしてしまいます。まったくの偶然がないわけではありませんが、たとえ偶然に良い音ができたとしても、それでは音を作ったことにならないでしょう。

さて、まず、音創りの基本手順をひととおり追ってみましょう。

○　**基本音色作り**

・アルゴリズムを決める
・マルチプルを決める
・アウトプットレベルとフィードバックを決める

○　**全体の音色作り**

・エンベロープを決める
・スケーリングを設定する
・データの調整

○　**エフェクトづけ**

・ＬＦＯとデチューンの設定

とくにこの手順にこだわる必要はなく、自分のやりやすい方法でかまわないのですが、この後の節ではこの手順にしたがって話を進めます。内容的には第２章や４．１節と重複する部分が多くなりますが、楽器音をシュミレートするときの役に立つように、できるだけいろいろな楽器の特徴について書くようにしました。

4．2．1　アルゴリズムを決める

まず、一番使いやすいのは、なんといってもAl4です。キャリアとモジュレータの組が2組ありますので、作りたい音を2つに分解してそれぞれの組に分担させて緻密に仕上げたり、同じ音を作っておいてデチューンをかけて厚みをつけたり、もちろんまったく違う音を同時に出させるというようなこともできます。特にデチューンに関しては、もっとも効果的な”斜めにデチューンをかける方法”を使うことができます。要するにオールマイティなアルゴリズムで、ＰＣのプリセットボイスでも圧倒的に多く使われています。

Al4につづいて使いやすいのが、Al5です。フィードバック付のモジュレータがいっぺんに3つのキャリアに変調をかけるわけで、事実上オペレータが6つあることになるという、いわばお徳用アルゴリズムです。3つものキャリアがあるため音の厚みを作りやすく、また、それぞれに変調がフィードバックも含めて2段にかかっていますから明るい派手な音を作ることができ、ブラス系の音には最適です。モジュレータがひとつしかないため、3つのキャリアとも似たような音になってしまう、という難点はありますが、各キャリアのマルチプルを変えるとか、エンベロープを変えるなど、工夫してください。似たような音をまとめてアンサンブル感を出すのに適したアルゴリズムといえるでしょう。

Al7は、キャリアはたくさんありますが、モジュレータが少ないため、あまり派手な音は作れません。マルチプルを全部変えてオルガン系の音を作るのに適しています。

Al0～Al3は、ひとつのキャリアに3つのモジュレータがよってたかって変調をかけているアルゴリズムで、派手な音を作るのに向いています。ソロでフレーズをひくリードシンセのような音から、うまく作ればソロのストリングスもできます。ただ、音に厚みが出ないのが最大の難点です。

この4種のアルゴリズムどうしの違いはあまりはっきりしませんが、キャリアに複数のモジュレータがのる場合、直列にのるよりも並列にのるほうが派手な音が出やすい傾向があります。ソロのストリングスなど比較的おちついた音にはAl0が良いでしょう。

4．2．2　マルチプルを決める

キャリアのマルチプルは、ベースの音など特に低い音や、特に高い音を作りたいということがないかぎり、1にしておきます。1オクターブ違う音は、周波数が2倍違うようになっていますので、1以外の場合でも、0.5、2、4、8のように設定して、オクターブ違いの音にするのが使いやすいでしょう。

AI0

30 EBASS 1
31 EBASS 2
43 KOTO
50 HARMONICA
70 FUNC BASS
72 METAL CLES

AI1

29 GUITAR
48 HARP
52 TIMPANI

AI2

17 GRANPRI
21 BRASS 1
22 BRASS 2
23 TRUMPET
24 STRING 1
25 STRING 2
28 EPIANO 3
38 OBOE
44 ZITAR
46 HARPSIC
64 CELLO
65 LOW BRASS
66 WW ENSNBL
67 AC GUITAR
71 SYN LEAD
73 STAIN
75 HUMAN
76 WOOD BASS

AI3

18 LASER 1
36 FLUTE
39 CLARINET
45 CLAV
59 COW BELL
63 MUSIC BOX
69 FUNC PERC

AI4

13 PIANO
19 LASER 2
26 EPIANO 1
27 EPIANO 2
34 PORGAN 1
35 PORGAN 2
40 GROCKEN
41 VIBRPHN
42 XYLOPHN
47 BELL
49 BELL/BRASS
51 STEEL DRUM
53 TRAIN
56 RAIN DROP
57 HORN
58 SNARE DRUM

AI5

32 EORGAN 1
78 SPACY
80 OLD STRING

AI6

15 DESCENT
16 UFO
37 PICCOLO
54 AMBULAN
55 TWEET
77 CHIMES

AI7

14 MUSHI
20 SIN WAVE
33 EORGAN 2
79 OBOE/B-CLAR

60 PERC 1
61 PERC 2
62 PERC 3
68 FLUTE/HARP
74 CUBIN GRASS
81 STEEL'S CRY

オルガンやブラスアンサンブルを、キャリアが複数のアルゴリズムで作る場合、各キャリアのマルチプルを異なる値にすることがあります。このような場合も0.5、2などがよく使われます。ただし、5度上の音も比較的よく混りあいますので、3、6などの値を設定することもないわけではありません。

モジュレータのマルチプルについても、(キャリアが1であれば)1がもっとも普通の設定です。FM音源によって得られる周波数は、

$$f = |fc \pm n \cdot fm|$$

で表わされますから、fc:fm = 1:1であればfcとともに、2 fc、3 fc、4 fc…、というようにすべての整数次倍音が含まれることになります。この音はくせがないので、さまざまな音色に幅広く使われます。特にブラスやストリングスには適しています。ちなみにアナログシンセサイザの音源としてよく使われるノコギリ波もすべての整数次倍音を含む音です。

さて、モジュレータのマルチプルを2にすると、3 fc、5 fc、7 fc、というように、奇数次倍音のみが発生します。奇数次倍音のみを含む音の典型としては矩形波が挙げられます。このような音は比較的素朴なおちついた音で、木管楽器に最適です。

3にすると、fc、2 fc、4 fc、5 fc、と、3の倍数次倍音が抑えられた形になります。これは弦をはじく音に近く、ギター、ウクレレ、琴などに使われます。

4のときは、2のときと同様に奇数次倍音のみを発生しますが、2のときよりも高次の倍音が増えた音になります。用途は2のときと同じと考えてよいでしょう。

以下、モジュレータのマルチプルを5、6、7と大きくするにしたがって、高次の倍音が増え、2 fc、3 fcなど低次の倍音が消えていきますので、ギラギラした、金属的な、くせの強い音になります。用途としては、クラビネット、ハープシコード、鐘の音などの他、エレピのアタック音など、音の一部として隠し味のように使うこともあります。(もちろんこの場合は、他にメインとなるキャリアがあるわけです。)

モジュレータのマルチプルを大きくすると、うまくすればFM音源ならではの変わった感じの音を作ることができますが、場合によっては、単にグシャグシャした、下品な音になりかねません。10以上の値を使うことはまずないと思ってください。とりあえず、1:1、1:2、あたりから慣れていくのがよいでしょう。なお、ここではキャリアのマルチプルが1であるとして話をすすめました。キャリアのマルチプルが2のときは、奇数次倍音を得るには4、8などを、3倍数次倍音を除くには6を、というように設定します。要するに、キャリアのマルチプルに対するモジュレータのマルチプルの比によって決まるのです。

4．2．3　フィードバックを決める

フィードバックは第２章でも述べたように、モジュレータのアウトプットレベルに相当するものです。ですから、特に地味な音、おとなしい音がほしいときや、フィードバック付きのオペレータがキャリアとなるAl７の場合を除いて、たいていの場合はひとつ増えたモジュレータを有効に使うために、大きな値に設定します。

もし音がギラギラしすぎるようなら、フィードバックではなく、アウトプットレベルを下げて落ち着けます。つまり、Sens と Depth の関係と同じように、フィードバックでおおまかに決めてアウトプットレベルで調節すると考えるのがいいでしょう。とにかく、フィードバックとそのオペレータのアウトプットレベルは常に１対のものとして、バランスをとるようにしてください。

4．2．4　アウトプットレベルを決める

キャリアのアウトプットレベルは常に０にしておきます。キャリアが複数ある場合に音量のバランスをとるため、メインとなるキャリアを０に、他のキャリアを小さく（設定値としては大きく）設定することもありますが、そのような場合以外は０にしておいてかまいません。

というのは、キャリアのアウトプットレベルは音色には基本的に関係ないからです。単に音量を調節するだけのものですから、ボイスデータに含める必要はありません。音量は、演奏時に他の方法で調節するべきです。

それに対し、モジュレータのアウトプットレベルは非常に重要です。これとエンベロープによって、FM音源の音色が決まると言っても過言ではありません。どれかひとつプリセットボイスをとって、モジュレータのアウトプットレベルを変えながら聞き較べてみてください。音色が明らかに変化するのがわかるでしょう。

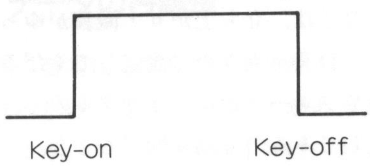
アウトプットレベルは対数的に変化し、設定値を８～９大きくすると、音量はほぼ1/2になります。さらにもう８～９大きくすると1/4になることになります。というわけですから、30～40をこえるとほとんど意味がなくなります。キャリアであれば音がほとんど聞こえない状態に、モジュレータであれば変調がほとんどかからない状態になります。また、このような状態では設定値を大きく変化させても音にはほとんど変化がありません。逆に小さな値ではわずかな値の変化が音に大きな影響を与えることになります。

というわけですので、モジュレータのアウトプットレベルはとりあえず20ぐらいに設定してみて、さらに明るく派手な音がほしければ少しずつ値を小さくしてみる、そうでなければ少しずつ大きくする、というようにして決めるのがよいでしょう。フィードバックの値とのかねあいも忘れないでください。また、エンベロープによる自然な音色変化を出したい場合は、モジュレータのアウトプットレベルを控えめにしてください。

4. 2. 5 エンベロープを決める

何度も述べてきたように、エンベロープはきわめて重要なパラメータです。極論すれば、これまでのパラメータ設定がどうあろうと、エンベロープ設定しだいで、どんな楽器に似た音もできるのです。エンベロープについてはこれまでにも述べてきましたので、ここでは各論的にいろいろな楽器のエンベロープの特徴を述べることにします。

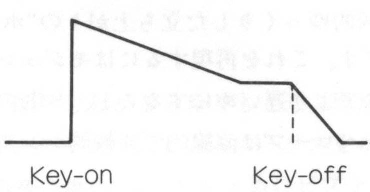
○オルガン



AR=31
DR=任意
SR=0
RR=31
SR=0

もっとも簡単なもののひとつです。Key-onと同時に音量が最大になる音ですから、ARは最大です。減衰もほとんどないので、SL=0、SR=0、またKey-offと同時に音は出なくなりますから、RRも最大です。音色変化もほとんどありませんから、キャリアもモジュレータも同じエンベロープパターンでよいでしょう。キークリック音を作っているオペレータがあるときは、DR、SLを大きくしてすぐに減衰するようにしてください。

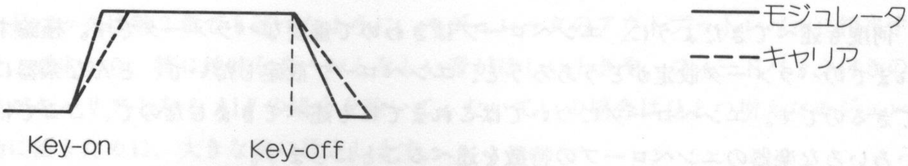
○エレピ



AR=31
SR=0
SR=7

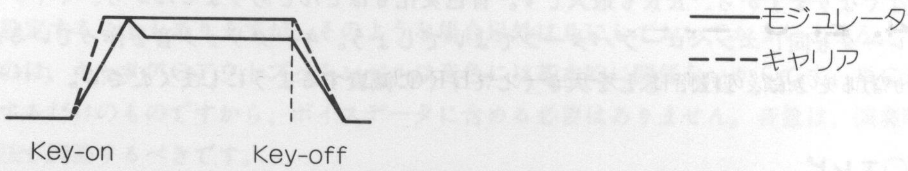
これもKey-onと同時に最大になりますから、ARは最大です。その後ゆるやかに減衰しますから、DR、SLを適当に調整します。上に挙げた値はめやすですから、自分の好みにあわせて調整してください。ただしリリースは短め（RRの数値は大きめ）のほうがいいと思います。音色・音量は同じような変化をしますから、キャリアとモジュレータと同じパターンでいいでしょう。

○ストリングス



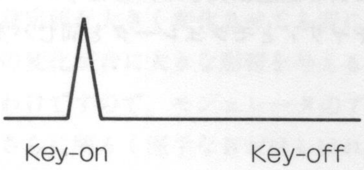
ストリングスのエンベロープの特徴はキャリアにあります。まず、ストリングスらしいゆるやかな立ち上がりと減衰を出すために、ARとRRを小さく設定することです。音色変化もほとんどありませんから、モジュレータもほぼ同じか、キャリアより立ち上がりを速く減衰を遅くします。ソロで細いフレーズをひくときなどは、立ち上がり・減衰がゆっくりすぎるとフレーズがグシャグシャになりますので、AR、DRを大きめに設定してください。また、その場合、弾き始めに倍音を多くすることによってフレーズがはっきりするので、モジュレータのSLを少し大きくして、DRを遅いめに設定してみるのも効果的でしょう。

○ブラス



ブラスの特徴は、音色変化を伴う比較的ゆっくりした立ち上がりの”ホワン”とした感じと、その後少し減衰してから安定することです。これを再現するにはモジュレータのエンベロープを調整します。まず、アタックをキャリアより遅いめにすること、SLの値をキャリアよりやや大きくすることです。キャリアのエンベロープは直線的で比較的シンプルです。リリースは短めにしておいてください。

○打楽器



AR＝31
DR＝任意／20
SR＝20／任意
RR＝10
SL＝0／15

打楽器の最大の特徴は、Key-on と同時に最大になり、Key-off にかかわらずすぐに減衰することです。

4．2．6　スケーリングを設定する

スケーリングはピアノ系の音をはじめ、ギター、琴などの打弦楽器、それに打楽器などにも有効です。全オペレータ、あるいはメインとなるオペレータのＫＳの値を変えてみてください。スケーリングは他のパラメータに較べて比較的音に対する影響が小さいので、設定するときにはまず最大値の３にしてみて、もし大きすぎると感じられたら小さくする、ということでよいでしょう。３以外の値を使うことはまずないと思います。

4．2．7　データの再調整

ここまでで音色の要素は完成していますが、エンベロープの設定によって最初に作った基本音色とはかなり違った音になっていることがありますので、ここでもういちど音色データを手直ししておきましょう。プリセットボイスのエディト例を参考にしながら、主にアウトプットレベルとフィードバックの値を修正してみてください。

4．2．8　エフェクトづけ

音創りの最後の段階は、デチューンやＬＦＯによってコーラス感などを出して音のリアルさを増すためのエフェクトづけです。それぞれのパラメータについてはこれまでにかなり述べてありますので、エンベロープと同様に、各楽器の特徴について述べることにします。

○エレピ

ピアノ系の音は、聞いてわかるとおりもともと周期的変化の少ない音ですから、ＬＦＯによる深い変調は用いず、デチューンによるコーラスをかける程度にするのが普通です。しかし、ローズピアノなどの音の場合、ごくひかえめにビブラート、トレモロをかけることもあります。

○オルガン

ピアノ系とは違い、デチューンを最大に利かせ、ＬＦＯも大いに使える音です。ビブラートをかけたり、ワウワウ、トレモロを組み合わせたりして、オルガンをレスリースピーカーにつないだようなロータリーサウンドを作ったり、最大限に活用してください。

○ブラス・ストリングス

これもエフェクトが欠かせない音です。ブラスやストリングスの音は、どんなに上手なプレイヤーの演奏でも、必ずピッチが揺れています。逆にピッチの揺れがない、平坦な音ではリアルさがありません。

ビブラートをさりげなくかけましょう。特に、ストリングスのソロでフレーズをとるような場合は、深めのビブラートが効果的です。また、ブラスのように音が明るくなると、どうしても厚みに欠けるように感じられますので、デチューンで厚みを出しコーラス感を与えることを忘れないでください。

○打楽器・ギター・クラビネット・ハープシコードなど

このような楽器は、厚みを出す必要はあまりありません。むしろ、厚みを出そうとすると音が丸くなってしまいますから、ギターらしい、クラビネットらしい硬い音を出すために、デチューンはかけないのが普通です。また、皮なり、弦なりが、叩いたあと大きく揺れてピッチ変化が起きますから、ビブラート、トレモロなどを適度にかけるとよいでしょう。

まとめてみましょう。まず打楽器系以外であれば、デチューンは必要と考えてください。ＬＦＯはビブラートがもっとも頻繁に使われます。次にトレモロ、ワウワウとなります。ＬＦＯの波形はほとんど三角波です。

効果音を作る場合には、ＬＦＯの波形はサンプル＆ホールドを用い、かなり速めのＰＭや、ノコギリ波のＡＭなどが役に立つでしょう。

4．2．9　EGミスセッティングの対策

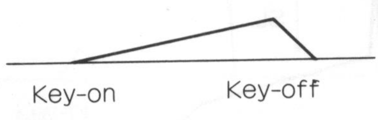
FM音源にはパラメータがたくさんありますので、操作に慣れないうちはパラメータの設定ミスがよく起こります。特に、ＥＧについては５個のパラメータがまとまってはじめてひとつのエンベロープを作るのでミスセッティングが起こりやすく、また、音が出なくなってしまうことがあるのでやっかいです。

ここでは、音が出てこない場合に考えられる、ＥＧのミスセッティングの例と対策について述べることにします。

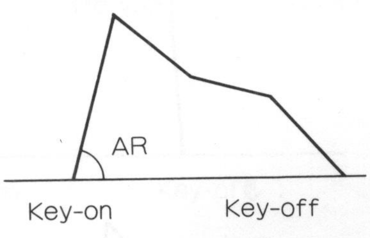
○ＡＲが小さすぎる

ＡＲが小さすぎると、Key-on 後しばらく音が出てきません。（0にすると永久に出てきません。）　細かいフレーズを弾いたりすると、Key-on 後音が立ち上がらないうちに Key-off になるため、音が出ないということになります。

――　対策　――　　ＡＲを大きくする。



⇒



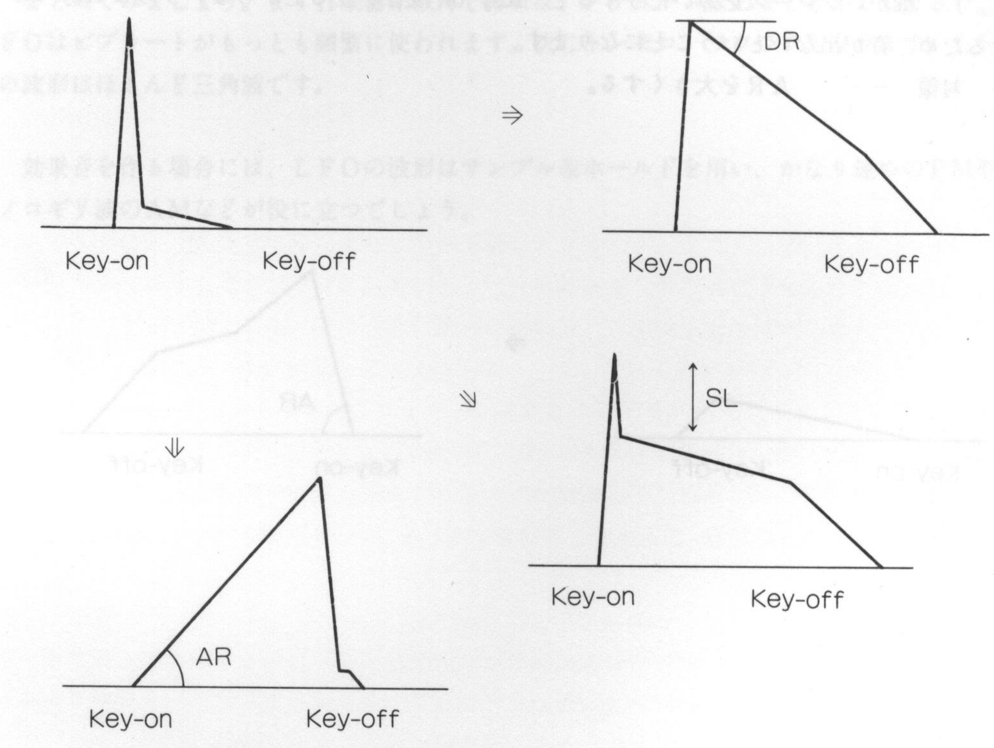
○ＡＲ・ＤＲ・ＳＬが大きすぎる

音はすぐに立ち上がりますがＤＲが大きいためすぐに減衰してしまいます。また、ＳＬが大きいために減衰量が大きく、実際に出力される音が小さくなってしまいます。ＳＬは対数的に減少することに注意してください。

—— 対策 ——　① ＤＲを小さくする

② ＳＲが比較的小さいとき、ＳＬを小さくするのも効果がある

③ ＡＲを小さくする



○ＡＲ・ＳＲが大きすぎ、ＳＬが小さすぎる

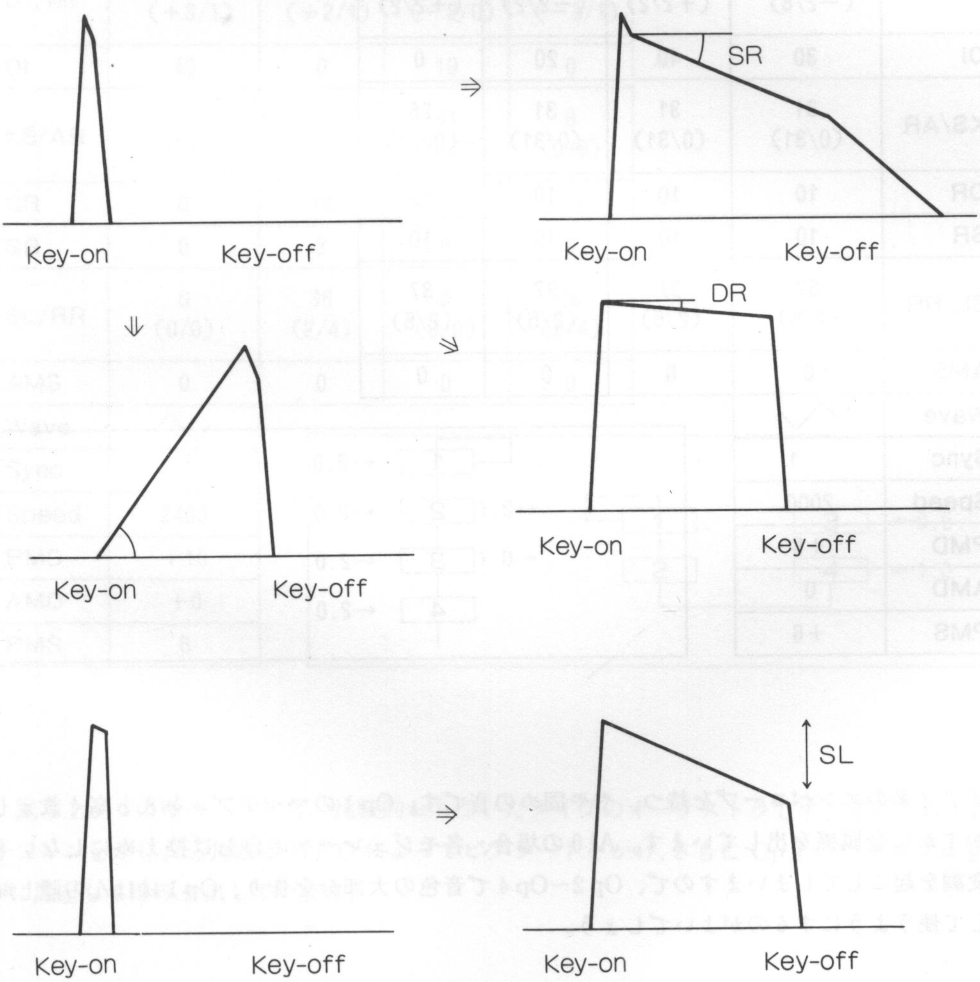
ＡＲが大きく、ＳＬが小さいので、Key-on後すぐにサスティン状態になりますが、ＳＲが大きいため、減衰してしまい音が出ないというものです。

－－　対策　－－　①　ＳＬを小さくする

②　ＤＲを小さくする

③　ＡＲを小さくする

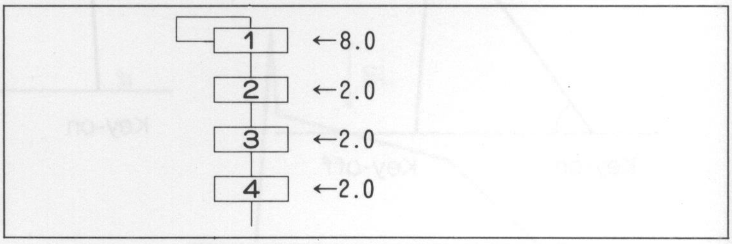
④　ＤＲが小さいなら、ＳＬを大きくするのも効果がある



4.3 オリジナル　サウンドデータ

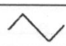
Synthe Piano	シンセピアノ	oct2～5

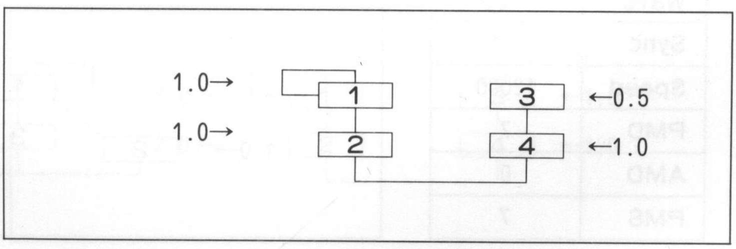
	Op1	Op2 3	Op3	Op4
Fb/Al	57 (7/0)			
Dt/Ml	104 (−2/8)	34 (+2/2)	98 (−2/2)	34 (+2/2)
Ol	30	40	20	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	25 (0/31)
DR	10	10	10	10
SR	10	10	10	10
SL/RR	37 (2/5)	37 (2/5)	37 (2/5)	37 (2/5)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	2000			
PMD	+8			
AMD	0			
PMS	+6			



ピアノ系のエンベロープを持つ、やや固めの音です。Op 1のマルチプルを8と高く設定してかすかに金属感を出しています。Al 0の場合、各モジュレータのＯＬは控えめにしないと過変調を起こしてしまいますので、Op 2～Op 4で音色の大部分を作り、Op 1はほんの隠し味として使うようにするのがよいでしょう。

Orchestre	オーケストラ	oct1〜7

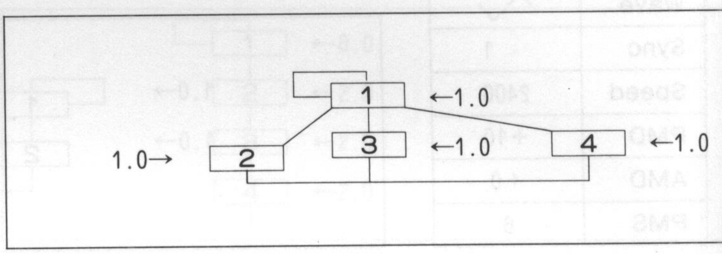
	Op1	Op2	Op3	Op4
Fb/AI		60 (7/4)		
Dt/MI	49 (+3/1)	33 (+2/1)	96 (−2/0)	113 (−3/1)
Ol	40	0	19	0
KS/AR	31 (0/31)	8 (0/8)	31 (0/31)	8 (0/8)
DR	0	13	0	13
SR	0	3	0	3
SL/RR	0 (0/0)	36 (2/4)	0 (0/0)	36 (2/4)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	2400			
PMD	+10			
AMD	+0			
PMS	6			



立ち上がりのゆっくりした、比較的おちついたタイプのオーケストラです。定石どおりにデチューンをかけ、三角波のＬＦＯによってビブラートをかけ、さらにOp3のマルチプルを0.5に設定して厚みを出しています。

	Original Brass1	軽快なブラス	oct3〜6

	Op1	Op2	Op3	Op4
Fb/AI	61 (7/5)			
Dt/MI	1 (0/1)	33 (+2/1)	81 (−1/1)	17 (+1/1)
OI	26	0	0	0
KS/AR	20 (0/20)	31 (0/31)	31 (0/31)	31 (0/31)
DR	10	12	12	12
SR	0	9	9	9
SL/RR	59 (3/11)	27 (1/11)	27 (1/11)	27 (1/11)
AMS	0	0	0	0
Wave				
Sync	1			
Speed	12000			
PMD	7			
AMD	0			
PMS	7			

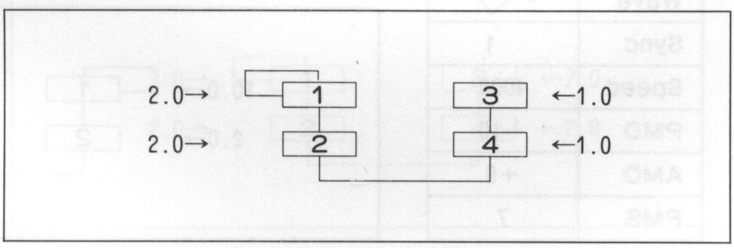


軽快で明るく歯切れよいブラスサウンドです。Op 2〜Op 4 のエンベロープを同じに設定し、デチューンでコーラス感を出し、PMDを使ってブラスらしさを出しています。

速いフレーズに使うと効果的でしょう。

Original Brass 2　　シンプルなブラス　　oct2〜5

	Op1	Op2	Op3	Op4
Fb/Al	60 (7/4)			
Dt/Ml	50 (+3/2)	34 (+2/2)	97 (−2/1)	113 (−3/1)
Ol	27	0	20	0
KS/AR	16 (0/16)	25 (0/25)	31 (0/31)	31 (0/31)
DR	10	10	31	31
SR	10	10	15	15
SL/RR	10 (0/10)	7 (0/7)	10 (0/10)	7 (0/7)
AMS	0	0	0	0
Wave	0			
Sync	1			
Speed	0			
PMD	0			
AMD	0			
PMS	0			

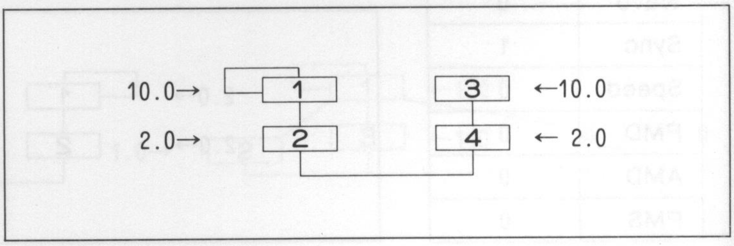


シンプルなブラスの音です。Op 1-Op 2 でメインの音を作り、Op 3-Op 4 で立ち上がりの音を作ってあります。Brass 1 よりはややおとなしく、ぬけのいい音といえるでしょう。

ＬＦＯのエフェクトは、あえて使わないでおきました。

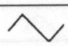
Kira Kira	星くずの音	oct3〜6

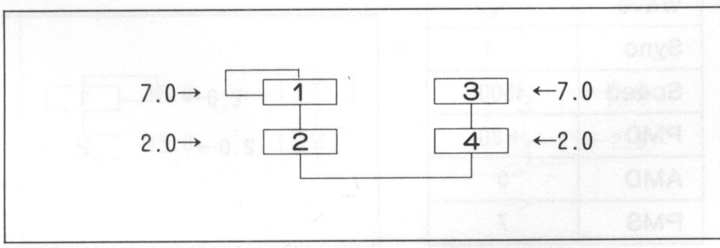
	Op1	Op2	Op3	Op4
Fb/Al	60（7/4）			
Dt/Ml	90 （−1/10）	98 （−2/2）	42 （+2/10）	18 （+1/2）
Ol	30	0	20	0
KS/AR	31 （0/31）	30 （0/30）	31 （0/31）	30 （0/30）
DR	12	15	12	15
SR	11	11	11	11
SL/RR	52 （3/4）	102 （6/6）	52 （3/4）	102 （6/6）
AMS	0	0	0	0
Wave	∿			
Sync	1			
Speed	4000			
PMD	+10			
AMD	+0			
PMS	7			



アメリカの人気ロックグループ"ＴＯＴＯ"の曲によく出てくるカランコロンとした金属的な音をイメージして作ってみました。モジュレータのマルチプルを10と、かなり高く設定して金属的な響きのもとになる独特の倍音を作っています。ＦＭ音源ならではの音といえるでしょう。このような音の場合、デチューンは少し控えめにするのがよいでしょう。

Chime　　かわいいドア・チャイム　　oct4〜6

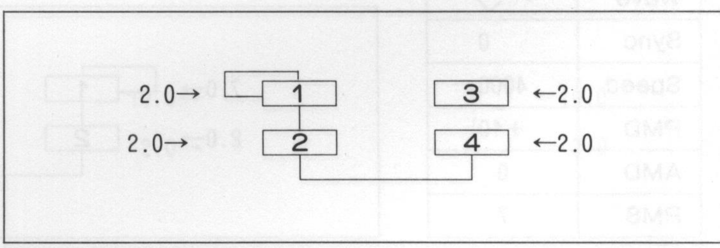
	Op1	Op2	Op3	Op4
Fb/Al	60 (7/4)			
Dt/Ml	119 (−3/7)	82 (−1/2)	23 (+1/7)	50 (+3/2)
Ol	30	0	20	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	10	10	10	10
SR	8	8	8	8
SL/R R	36 (2/4)	36 (2/4)	36 (2/4)	36 (2/4)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	4000			
PMD	+10			
AMD	0			
PMS	7			



この音色データの特徴は、マルチプルにあります。ｃ：ｍ＝２：７という設定で非整数次倍音を発生させ、金属感のある鐘の音を作り出しています。ＥＧはピアノ型で、ＬＦＯでビブラートをかけています。ビブラートはKey-on、Key-offの時間しかかかりませんので、扱いには注意してください。

Voice	仙人のこえ	oct2〜6

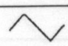
	Op1	Op2	Op3	Op4
Fb/Al	4 (0/4)			
Dt/MI	18 (+3/2)	34 (+2/2)	98 (−2/2)	114 (−3/2)
Ol	35	0	35	0
KS/AR	18 (0/18)	18 (0/18)	18 (0/18)	18 (0/18)
DR	0	0	0	0
SR	0	0	0	0
SL/RR	6 (0/6)	6 (0/6)	6 (0/6)	6 (0/6)
AMS	0	0	0	0
Wave	∿			
Sync	1			
Speed	4500			
PMD	+20			
AMD	0			
PMS	7			

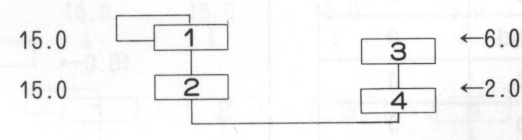


幽玄な感じのヒューマンボイスです。このような音はフィードバックおよびモジュレータのＯＬを控えめにして過変調にならないようにすることが第一条件です。また、立ち上がり、立ち下がりが急になりすぎないようにＥＧを設定し、デチューンとＬＦＯでエフェクトをかければでき上がりです。

Simons	シンセドラム	oct3〜6

	Op1	Op2	Op3	Op4
Fb/Al	52 (6/4)			
Dt/Ml	15 (+0/15)	15 (+0/15)	86 (−1/6)	18 (+1/2)
Ol	0	0	20	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	1	16	22	16
SR	0	16	6	16
SL/RR	37 (2/5)	248 (15/8)	105 (6/9)	248 (15/8)
AMS	0	0	0	0

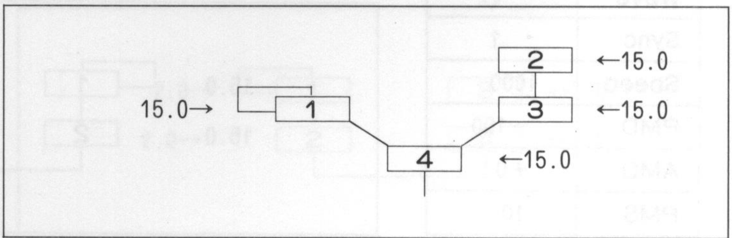
Wave	
Sync	1
Speed	1000
PMD	−100
AMD	+0
PMS	10



いわゆるシンセドラムの音です。Op 1-Op 2で叩いた瞬間の打撃音ノイズを作り、Op 3-Op 4で「ピョーン」という感じにピッチが下がっていく部分を作っています。PMDをマイナスにしてひっくりかえしたノコギリ波によるピッチ変化をうまく利用した例ですが、Speedとキャリアの EGをうまく調整しないと、LFOの立ち上がり部分の音まで出てしまいます。Op 3のマルチプル、OLは好みの音になるよう調整してください。

Bomb	どっかーん	oct1〜8

	Op1	Op2	Op3	Op4
Fb/Al	58 (7/2)			
Dt/Ml	15 (0/15)	15 (0/15)	15 (0/15)	15 (0/15)
Ol	0	0	0	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	0	0	0	31
SR	0	0	0	8
SL/RR	0 (0/0)	0 (0/0)	0 (0/0)	4 (0/4)
AMS	0	0	0	0
Wave				
Sync	0			
Speed	0			
PMD	0			
AMD	0			
PMS	0			

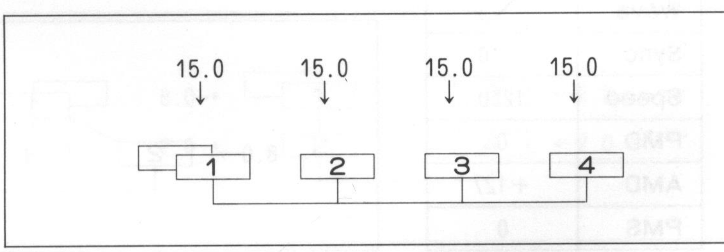


単なる爆発音です。各オペレータの出力を最大に、フィードバックも最大にすると、出てくる音はもはやノイズでしかありません。わずかに音程感がのこりますが、マルチプルを最大にするとそれも消えて、音程にかかわらず同じ音が出るようになります。そこで、キャリアのエンベロープを調節して、爆発音風の音量変化を出してみました。

作るのが簡単なわりにはよくできた音色です。

Seaside	渚にて	oct

	Op1	Op2	Op3	Op4
Fb/Al	58(7/2)			
Dt/MI	15 (+0/15)	15 (+0/15)	15 (+0/15)	15 (+0/15)
Ol	0	0	0	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	0	0	0	0
SR	0	0	0	0
SL/RR	0 (0/0)	0 (0/0)	0 (0/0)	4 (0/4)
AMS	0	0	0	0
Wave	/\/			
Sync	1			
Speed	150			
PMD	0			
AMD	+10			
PMS	0			



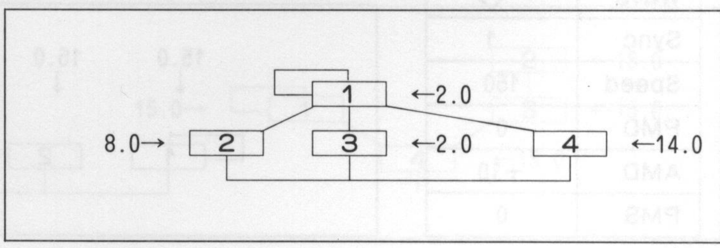
ＬＦＯのＡＭを使った典型的なＳＥの例です。基本的にはＢｏｍｂを少しエディトしたものです。ＥＧを持続音型にして、キャリアにリリースを少しかけ、三角波でＬＦＯをキャリアのＡＭにかけています。

波の満ち引きが周期的すぎるのが玉にきずですが、よくできたといえるでしょう。

この音の、Speed を 10383、Wave をＳ＆Ｈ、ＡＭＤを 30 とすると、線香花火のような音になります。

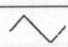
Emergency Call	緊急事態発生！	oct4

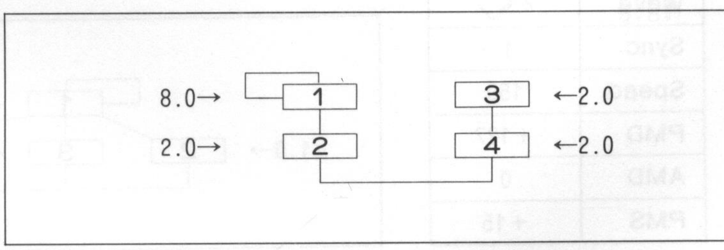
	Op1	Op2	Op3	Op4
Fb/Al	61 (7/5)			
Dt/MI	50 (+3/2)	120 (−3/8)	50 (+3/2)	126 (−3/14)
Ol	20	0	0	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	0	0	0	0
SR	0	0	0	0
SL/RR	15 (0/15)	15 (0/15)	15 (0/15)	15 (0/15)
AMS	0	11	11	11
Wave	/\/			
Sync	0			
Speed	1250			
PMD	0			
AMD	+127			
PMS	0			



これもＬＦＯのAMを使ったＳＥの例です。フィードバックをかけたモジュレータとマルチプルを高くしたキャリアの組み合わせで、ブザーらしい音を作り、デチューンでうねりを与え、矩形波のＬＦＯで断続音にしています。Speedを250ぐらいにすると、電話のベルに似た音になります。

Spacy	2001年宇宙の旅	oct

	Op1	Op2	Op3	Op4
Fb/AI	60 (7/4)			
Dt/MI	120 (−3/8)	98 (−2/2)	34 (+2/2)	50 (+3/2)
OI	30	0	20	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	10	10	10	10
SR	7	7	7	7
SL/RR	41 (2/9)	41 (2/9)	41 (2/9)	41 (2/9)
AMS	0	0	0	0
Wave				
Sync	ON			
Speed	1			
PMD	+88			
AMD	0			
PMS	9			

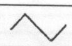


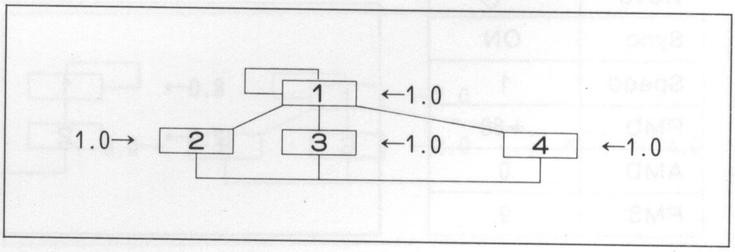
ＬＦＯのＰＭを使ったＳＥの例です。約１オクターブぐらいの範囲で音程がゆっくり上下しながら、ＥＧの設定によって音量が小さくなっていきます。

私は「2001年宇宙の旅」のようなＳＦ映画のシーンをイメージしながら作りましたが、どうでしょうか。SpeedやＰＭＤは好みにあわせて調節してください。

TV GAME	テレビゲームサウンド	oct

	Op1	Op2	Op3	Op4
Fb/Al	61 (7/5)			
Dt/MI	1 (+0/1)	49 (+3/1)	1 (+0/1)	49 (+3/1)
Ol	25	0	0	0
KS/AR	31 (0/31)	31 (0/31)	31 (0/31)	31 (0/31)
DR	0	0	0	0
SR	10	10	10	10
SL/RR	10 (0/10)	10 (0/10)	10 (0/10)	10 (0/10)
AMS	0	0	0	0

Wave	
Sync	1
Speed	150
PMD	+127
AMD	0
PMS	+15



テレビゲームやＳＦアニメの効果音としてよく出てきそうな音です。この音の最大の特徴は、ＰＭＤ、ＰＭＳを大きくしたＬＦＯ効果の面白さにあり、そのかわり、各オペレータのパラメータはかなり単純です。マルチプルをいじるとちょっと変わった感じになるので、調整してみてください。

4.4 ＤＸユーザへの指針

読者のみなさんのなかには、ヤマハのＤＸシリーズなどのシンセサイザをとおしてＦＭ音源に触れていた方、また実際に使用されている方もいることと思います。ＤＸに内蔵されている音をＰＣに、あるいは逆にＰＣに内蔵されている音をＤＸに移植したいと考えている方も多いことでしょう。

ここではそのような方の役に立つことをいくつか解説します。とりあげる機種は、普及度のもっとも高いＤＸ７です。

ＤＸとＰＣに搭載されているＦＭ音源は、基本的に同じものですが、そのままデータが移せるわけではありません。ＤＸにあってＰＣにないパラメータがいくつかあります。また同じパラメータであっても、値の範囲が異なる場合がほとんどです。

まず、両者のパラメータの対応を図に示しておきます。

－－－－－－－－－－－－－－－－－－－－－－ひとことメモ－－－－－－－－－－－－－－－－－－－－－－

ヤマハのシンセサイザＤＸシリーズは、ＦＭ音源を心臓部として、さらに演奏に必要なさまざまな付加機能をもつシンセサイザです。１９８３年春の発売いらい、プロにも満足のいく性能と、アマチュアにも手の届く価格とで、国内はもとより海外でもベストセラーとなっています。ＴＯＴＯ、カシオペアなど、多くの著名なミュージシャンにも愛用されています。

－－－

DX7		PC	
パラメータ名称	可変範囲	パラメータ名称	可変範囲
Algorithm	32種	←	8種
Feedback	0→7	←	0→7
LFO Wave	6種	←	6種
Speed	0→99	←	0→16383
Delay	0→99	Sync Delay—time	2～255
PMD	0→99	←	−127→127
AMD	0→99	←	−127→127
Sync	On-Off	←	On—Off（ディレイあり
PMS	0→7	←	0→15
AMS	0→3	←	0→15
MODE/SYNC	Ratio-Fixed / On-Off	——	——
Freguency Coarce	0.5→31	Multiple	0→15
Fine	×1→1.99	——	——
Detune	−7→7	←	−4→3
EG Rate1	0→99	Attack Rate	0→31
Rate2	0→99	Decay Rate	0→31
Rate3	0→99	Sustain Rate	0→31
Rate4	0→99	Release Rate	0→15
Level1	0→99	——	——
Level2	0→99	Sustain Level	15→0
Level3	0→99	——	——
Level4	0→99	——	——
Level Sealing関係		——	——
Rate Scaling	0→7	←	0→3
Output Level	0→99	←	127→0
Key Velocity	0→7	——	——
Pitch EG		——	——

たとえばフィードバックはどちらも可変範囲は０～７でみかけのうえでは同じですが、ＤＸ側の７とＰＣ側の７が、同じフィードバックの程度を表わすとは限りません。他のパラメータについても同じことがいえますので、ＤＸのデータを正確に換算してＰＣに移植することは、

厳密にはまず不可能です。

さしあたり妥協案として、最大値どうし、最小値どうしは同程度の効果を持つものと仮定し、その間の値については按分比例でおおまかに計算することにします。（たとえば、ＤＸでRate 1 = 50なら、ＰＣではAR = 16というように、比率で計算する。）

あとは耳を頼りに調整するしかないでしょう。

次に、各パラメータについて、詳しく考えてみます。

○ マルチプル

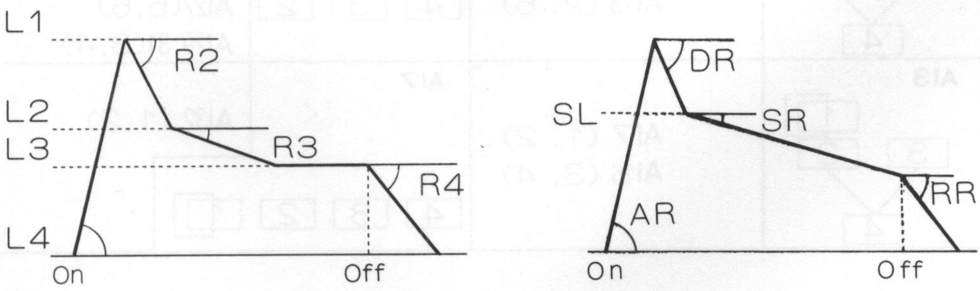
MultipleとFrequencyは、名称は違いますが、内容は同じで、オペレータの倍音構成を表わすパラメータです。このパラメータについては、前述の妥協案はもちろん不要です。つまり、Frequency = 4であれば、Multiple = 4のように、正確に対応します。（ただし、Multiple = 0は便宜上の値であって、この場合は0.5として計算されていますから、Frequency = 0.5に対応します。）

Multipleは15までしかありませんが、16以上の値を使うことはほとんどありませんから問題ないでしょう。しかし、ＰＣにはＤＸにおけるFrequency Fine、Fixed Modeがありませんから、それらによる非整数次倍音を用いた音色は再現できません。

○ ＥＧ

ＤＸの8パラメータに対して、ＰＣは5パラメータです。ＤＸのＲ1、Ｒ2、Ｒ3、Ｒ4、Ｌ2がそれぞれＳＲのAR、DR、SR、RR、SLに対応し、残りのパラメータは、Ｌ1 = 99、Ｌ3 = Ｌ4 = 0に固定されていると考えてください。

たいていの音色で、Ｌ1は99、Ｌ4は0に設定されていて問題ないのですが、Ｌ3は0でない場合が多いので、多少工夫が必要です。また、Ｌ3>Ｌ2となる、2つピークを持つような音色は、ＰＣでは作れません。



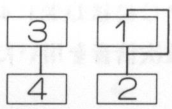
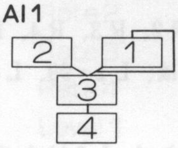
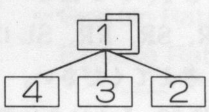
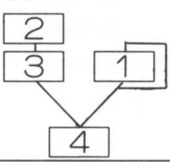
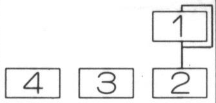
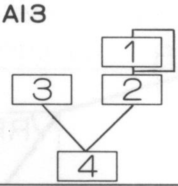
○　LFO

Waveは、DXにあってPCにないものが1つあります。 〜 です。〜 は、たいていの場合、╲╱ で代用します。

○　アルゴリズム

PCは4つのオペレータを持ち、アルゴリズムは8種類です。一方DX7は、オペレータは6つ、アルゴリズムは32種類です。そこで、DX7の音色をPCに移植する場合は、影響の少ないオペレータを2つマスクしてPCのアルゴリズムにあてはめることになります。

次にアルゴリズムの対応表を示します。オペレータの番号が、PCでは上→下の順に1→4となっていますが、DXでは逆に下→上の順に1→6となっていることに注意してください。

PC	DX7(削除Op)	PC	DX7(削除Op)
Al0 	Al1 (1, 2)	Al4 	Al2,8,9,5(5,6) Al5,29(1,2) Al11,19(3,5) Al12,13,22(4,5) Al20,21,23,26,27(1,5)
Al1 	Al14 (1, 2)	Al5 	Al22,24(1,2)
Al2 	Al8 (1, 2) Al17 (5, 6) Al18 (2, 6)	Al6 	Al5(2,4) Al21(2,6) Al23,24,25,26(3,5) Al27(5,6) Al29,31(3,4)
Al3 	Al7 (1, 2) Al16 (3, 4)	Al7 	Al32 (1,2)

表の見方について説明します。たとえばＰＣのAl 6は、ＤＸ７のAl 5でOp 1、Op 2をマスクしたもの、または、Al 23やAl 24などでOp 3、Op 5をマスクしたものに相当していることを示します。

さらに詳しくみると、次のことがわかります。ＤＸ７のAl 5、21、23、26、27などは、ＰＣのAl 4、Al 6、で表わせますが、Al 3、４、６、10、28、30は、どの２つのオペレータをマスクしてもＰＣのアルゴリズムにはあてはまりません。ただし、フィードバックが無視できる音色であれば、たとえばAl 3、Al 4はOp 3、Op 6をマスクして、ＰＣのAl 4にあてはめることができます。

なお、ＤＸ９、ＤＸ２１などは、オペレータ４つ、アルゴリズム８種で、ＰＣとまったく同じですから、対応させるのは簡単です。ＤＸでのAl 1～Al 8がＰＣでのAl 0～Al 7に対応すること、オペレータの番号が逆向きについていることに注意してください。

○　その他

Key Velocityは、鍵盤をたたく動作がない限り不要ですから、特に問題はありません。Pitch EG、Level Sealing関係はそれに相当するパラメータがないため、再現不可能です。これは、「FM音源の」ではなく、「DXの」持ち味となっているところですから、しかたがないともいえるでしょう。ただし、Pitch EGは簡単なものであれば、ＬＦＯのＰＭＤを利用して、同じような効果を得られますので試してみてください。

予想されたことではありますが、ＤＸ７の完成度の高い音を、オペレータの数もパラメータの種類も少ないＰＣで再現することは、かなりの困難を伴います。まったく同じ音を再現することはそもそも無理なのですから、むしろ新しい音をつくるつもりでとりくむのがよいと思います。

表の見方について説明します。たとえばＰＣのAl6は、ＤＸ７のAl5でOp1、Op2をフラットにしたか、または、Al23やAl24などでOp3、Op5をフラットにしたものに相当していることを示します。

さらに詳しくみると、次のことがわかります。ＤＸ７のAl5、21、23、26、27などは、ＰＣのAl4、Al6、で表わせますが、Al3、4、6、10、28、30は、どの[illegible]ることができます。

[illegible]ＤＸ７のAl1～Al18が[illegible]ＰＣの[illegible]、オペレータの番号が逆向きについていることに注意してください。

○ その他

Key Velocity は、[illegible]動作がない限り不要ですから、特に問題はありません。Pitch EG、Level Scaling 関係はそれに相当するパラメータがないため、再現不可能です。これは、「ＦＭ音源の」ではなく「ＤＸの」持ち味となっているところですから、しかたがないとあきらめましょう。ただし、Pitch EG は簡単なものであれば、ＬＦＯのＰＭＤを利用して、同じような効果を得られますので試してみてください。

予想されたことではありますが、ＤＸ７の完成度の高い音色はオペレータの数もアルゴリズムの種類も少ないＰＣで再現することは、かなりの困難を伴います。まったく同じ音を再現することはそもそも無理なのですから、むしろ新しい音をつくるつもりで[illegible]のがよいと思います。

Appendix

付　録

FM音源スーパーサウンド

付録A　音色データの格納形式

ＰＣには、あらかじめ６９種類の音色が内蔵されています。その詳細については３章で示したとおりですが、ここではそのデータがＲＯＭ上にどのような形式で格納されているか説明します。音色データを直接参照できると、のちほど説明するサウンドBIOSを使用する際に便利です。

音色データは、１つの音色に付き５０バイトで構成され、サウンドボード上の拡張ＲＯＭのアドレスＣＣ００Ｈ：１Ｅ０ＡＨからの領域に格納されています。また、それぞれの音色データについて、その位置を表すアドレステーブルがＣＣ００Ｈ：１Ｄ６４Ｈからの領域に格納されています。音色データは１３～８１の各音色に対応するものですが、アドレステーブルは0～８１の各音色に対応します。

CC00:1D64	音色番号０の格納されているオフセット
CC00:1D66	音色番号１の格納されているオフセット
	………
CC00:1E08	音色番号81の格納されているオフセット
CC00:1E0A	音色番号13の音色データ（50バイト）
CC00:1E3D	音色番号14の音色データ
	………
CC00:2BC9	音色データ81の音色データ

図5-1　音色データの格納形式

アドレスは、あくまでも目安です。ＲＯＭのバージョンによってはこれと異なることがあります。

オフセット	内　容			コメント
0		Feedback	Algorithm	
1〜4	Attack Rate			それぞれ，Op1,2,4にに対応する。
5	Operator Mask			
6〜9	Decay Rate			それぞれ，Op1,2,3,4に対応する。
10	LFO Wave			
11〜14	Sustain Rate			それぞれ，Op1,2,3,4に対応する。
15	LFO Sync Delay Time			
16〜19	Release Rate			それぞれ，Op1,2,3,4に対応する。
20〜21	LFO Speed			ワードデータ
22〜25	Sustain Level			それぞれ，Op1,2,3,4に対応する。
26	Pitch Modulation Depth			
27〜30	Output Level			それぞれ，Op1,2,3,4に対応する。
31	Amplitude Modulation Depth			
32〜35	Key Scaling			それぞれ，Op1,2,3,4に対応する。
36	Pitch Modulation Sense			
37〜40	Multiple			それぞれ，Op1,2,3,4に対応する。
41	Reserved			
42〜45	Detune Rate			それぞれ，Op1,2,3,4に対応する。
46	Reserved			
47〜50	Amplitude Modulation Sense			それぞれ，Op1,2,3,4に対応する。

図5-2　音色データの構造

付録B　サウンドBIOS

サウンド関連のステートメント／関数の機能を実現するためのソフトウェアとして、サウンドBIOSがユーザに開放されています。サウンドBIOSを用いることにより、ユーザの開発する機械語プログラムなどでの音楽演奏が容易になります。ここでは、サウンドBIOSについての事項を簡単に紹介します。

サウンドBIOSを使用する場合、MUSIC BIOSは使用できません。なぜなら、サウンドBIOSとMUSIC BIOSはボード上のROMで供給され、またこれらのボードは同時には実装できないからです。

B-1　サウンドインタフェース

サウンドBIOSとのインタフェースは、各種レジスタ、制御情報域、パラメータリストによって行われます。各種レジスタは、コマンドの種別や、データのありかを示すアドレスなどを表しますが、制御情報域やパラメータリストは、実際にメモリ上に領域がとられ、これはユーザが用意してあげなければなりません。

各種レジスタやパラメータリストについてはそれぞれのコマンドの項で解説しますが、すべてに共通する事項として、制御情報域についての解説を行っておきます。制御情報域は、256バイト（実際はそれに続く局所的作業域が含まれるために384バイト）の大きさがあり、それが以下のように区分されています。

○オフセット００H：ＰＬＡＹバッファのオフセットアドレス（チャンネル１）

ＰＬＡＹ文で使用する演奏用バッファのアドレスのうち、オフセットを表します。これは、BIOSのINITIALIZEコマンド実行時に、ユーザの用意するバッファのアドレスをセットしておく必要があります。

○オフセット０２H：ＰＬＡＹバッファのセグメントアドレス（チャンネル１）

オフセット００Hと同様です。

○オフセット０４H：ＰＬＡＹバッファのサイズ（チャンネル１）

ＰＬＡＹ文で使用する演奏用バッファのサイズをバイト数で入れます。初期化に関してはオフセット００Hと同様です。

○オフセット０６H：ＰＬＡＹバッファ内ポインタ（チャンネル１）

演奏中のデータへのポインタを保持します。BIOS内で使用されるものなので、初期化は必要ありません。

○オフセット０８Ｈ：ＰＬＡＹバッファ残量（チャンネル１）

ＰＬＡＹバッファ内の残量をバイト数で保持します。ここが０のとき、演奏用バッファはいっぱいであることを示しています。

○オフセット０ＡＨ：ＰＬＡＹバッファエンプティ割り込み条件（チャンネル１）

ＰＬＡＹバッファが空になった際の割り込みを行う際の条件を保持します。ビット１５が割り込みが可能か否かを示し、１で割り込み可、０で割り込み不可（未設定）であることを表しています。ビット１４〜０はビット１５が１であるときのみ有効で、割り込みを発生させるバイト数を表します。このバイト数の値にバッファの有効データ数が達したときに、割り込みが発せられることになります。

○オフセット０ＣＨ：エンプティ割り込みアドレス（チャンネル１）

バッファ残量がエンプティになったときの割り込み処理ルーチンのアドレスです。オフセット：セグメントの順に入っています。

○オフセット１０Ｈ：カレントＫＥＹ番号（チャンネル１）

現在発音中の音のＫＥＹ番号が、０〜６０Ｈの値で格納されています。値は、ＰＬＡＹ文のMMLのＫコマンドに対応したものです。値がこの範囲外にあるとき、発音されていません。

○オフセット１１Ｈ：デフォル肇音長（チャンネル１）

SET LENGTHコマンドによって設定された、デフォルトの音長が格納されています。

○オフセット１２Ｈ：デフォルトタッチ（チャンネル１）

SET TOUCHコマンドによって設定されたＧ／Ｓ値が格納されています。

○オフセット１３Ｈ：演奏中フラグ（チャンネル１）

該当するチャンネル（この場合チャンネル１）で演奏中であるかどうかを示しています。ＦＦＨで演奏されていて、０で演奏されていないことを表します。

○オフセット１４Ｈ〜１ＦＨ：サウンドワークエリア１（チャンネル１）

サウンドBIOSが使用する作業領域です。ユーザが使用することは禁じられています。

○オフセット２０Ｈ〜３ＦＨ：チャンネル２
○オフセット４０Ｈ〜５ＦＨ：チャンネル３

○オフセット６０Ｈ～７ＦＨ：チャンネル４
○オフセット８０Ｈ～９ＦＨ：チャンネル５
○オフセットＡ０Ｈ～ＢＦＨ：チャンネル６
チャンネル１と同様の作業領域が、各チャンネルごとに用意されています。

○オフセットＣ０Ｈ：テンポ数
SET TEMPO コマンドによって設定したテンポ数が格納されています。

○オフセットＣ１Ｈ：ＫＥＹステータスセーブエリア
ALL STOP コマンドによってセーブされるＫＥＹステータスが格納されています。

○オフセットＣ２Ｈ～ＦＦＨ　　：サウンドワークエリア２
○オフセット１００Ｈ～２ＦＦＨ：サウンドワークエリア３
サウンドBIOSが使用する作業領域です。ユーザが使用することは禁じられています。

制御情報域は、各コマンドにおいてレジスタESのみによってその位置が指定されますので、オフセットは強制的に0となります。

B-2 割り込みベクタの設定

サウンドBIOSは、内部割り込みによってその機能を呼び出します。これは、BASICを使用するとき以外はベクタが初期化されていませんので、ユーザが初期化を施してやる必要があります。このための情報が、アドレスＣＥＥ０：０に以下の形式で格納されています。

```
ＣＥＥ０：００００　０１Ｈ，？，？，？（？は不定であることを示す）
　　　　　０００４　Ｄ２Ｈ，００Ｈ
　　　　　０００６　サウンドBIOSエントリアドレスオフセット
```

ここで、サウンドBIOSのエントリアドレスは、セグメントＣＥＥ０からのオフセット値です。これを、適当な割り込みベクタにセグメント値とともに書き込んでやれば、ベクタの初期化は完了です。BASICSでは、ベクタ番号Ｄ２Ｈが採用されています。

B-3 共通事項

サウンドBIOSを使用するにあたって、いくつか共通事項があります。
サウンドBIOSの実行においては、スタックが９６バイト必要です。
サウンドBIOSにおいて、特に返り値の入らないレジスタは保存されます。

サウンドBIOSにおいて、演奏中でも外部割り込みは受け付けられます。

サウンドBIOSでは、2つのコマンド指定方式を選択できます。1つ目はリアルタイム機能で、実行要求後すぐに実行される方式です。2つ目はディレイド機能で、PLAYコマンドに与えるディレイド機能フォーマットのデータとして与えられ、演奏につれて実行される方式です。ディレイド機能はリアルタイム機能の一部を含み、両者は実行形態の違いや情報の受渡し方式を除けばほぼ同一機能となります。

B-4 各コマンド解説

○ INITIALIZE

機　能　　サウンドボード／BIOSの初期設定および制御情報域の設定

入　力　　AH←00H

　　　　　ES←制御情報域セグメント

出　力　　なし

サウンドボードとサウンドBIOSの初期設定を行い、同時に与えられた制御情報域も初期化します。本コマンドを除くサウンドBIOSコマンドを実行する際には、必ず本コマンドを事前に実行しておかなければなりません。また、制御情報域のオフセット00H，02H，03H（チャンネル1の場合、チャンネル2以降についても同様）には、必ずPLAYバッファのアドレスとサイズを設定しておかなければなりません。同時に、PLAYバッファを、使用するチャンネルの数だけ、用意しておく必要があります。この時、PLAYバッファは偶数サイズでなければなりません。

本コマンドの実行によって、以下のパラメータが初期化されます。

音長：	48ステップ
テンポ：	120（48ステップ／秒）
ゲートタイム／ステップタイム：	8

音源の音色パラメータは初期設定されないため、本コマンドの実行後に設定を行う必要があります。

○ PLAY

機　能　　与えられたデータ列（ディレイド機能）を実行する

入　力　　AH←01H

　　　　　ES:BX←パラメータリストのアドレス

出　力　　なし

本コマンドは、与えられるパラメータリストの指示に従って、演奏を開始します。パラメータリストは、各チャンネルに対応するデータブロックのアドレスをまとめた表で、次のような構成となっています。

	+0（1ワード）	+2（1ワード）
ES:BX+00H	データブロックセグメントベース	未使用
+04H	データブロック1　オフセット	データブロック1　データ長
+08H	データブロック2　オフセット	データブロック2　データ長
+0CH	データブロック3　オフセット	データブロック3　データ長
+10H	データブロック4　オフセット	データブロック4　データ長
+14H	データブロック5　オフセット	データブロック5　データ長
+18H	データブロック6　オフセット	データブロック6　データ長

図5-3　パラメータブロック

本コマンドは、与えられたディレイド機能フォーマットのデータをＰＬＡＹバッファへ転送するのみで、実際の演奏は別に用意される演奏ルーチンが、インターバルタイマの割り込みによって行います。データブロックは、INITIALIZEコマンド実行時に用意したＰＬＡＹバッファより短くなくてはなりません。また、データブロック内にはディレイド機能以外のデータがあってはいけません。もしそのような場合の動作は保障されません。

○CLEAR BUFFER

機　能　　現在の演奏を中止し、ＰＬＡＹバッファをクリアする

入　力　　AH←02H

　　　　　AL←処理指定（00H：演奏中止とバッファクリアを行う

　　　　　　　　　　　　01H：演奏中止、バッファクリア、制御情報域を初期化する）

出　力　なし

本コマンドは、現在の演奏を中止し、ＰＬＡＹバッファをクリアします。本コマンドは、サウンドBIOSの使用を終了する際には、必ず実行しなければなりません。

○ READ REGISTER

機　能　指定されたＯＰＮレジスタの内容を読み出す

入　力　AH←10H

　　　　AL←レジスタ番号

出　力　BH←00H

　　　　BL←設定されている値

本コマンドは、ＯＰＮのレジスタに設定されている内容を読み出すコマンドですが、ＯＰＮのレジスタ（特にFM部）は読み出し不能な場合が多いため、WRITE REGISTERコマンドによって書き出した内容のみが有効となります。

○ WRITE REGISTER

機　能　指定されたＯＰＮレジスタに値を書き込む

入　力　AH←11H

　　　　AL←レジスタ番号

　　　　BL←設定する値

出　力　なし

本コマンドは、ＯＰＮのレジスタに値を書き込むコマンドで、READ REGISTERコマンドを用いて値の読取りを行う際には、必ず本コマンドと対にしなければなりません。また、ＯＰＮのレジスタのうち、その後の正常の動作が期待されないような書き込み（インターバルタイマなど）は行ってはいけません。

本コマンドは、次に示すディレイド機能フォーマットでも実現できます。

　第1バイト：　８１H

　第2バイト：　レジスタ番号

　第3バイト：　設定する値

○ SET G/S

機　能　G／S値（ゲートタイム／ステップタイム）を設定する

入　力　AH←12H
　　　　AL←指定チャンネル
　　　　BL←G／S値
出　力　なし

本コマンドは、G／S値（音長に対しての実際の発音時間）を設定します。G／S値と実際の比率は、次のようになっています。

G／S値　00H　：　1／8
G／S値　01H　：　2／8
G／S値　02H　：　3／8
G／S値　03H　：　4／8
G／S値　04H　：　5／8
G／S値　05H　：　6／8
G／S値　06H　：　7／8
G／S値　07H　：　8／8

この時、S値（ステップタイム）が12以下の時には、期待したとおりの分割とはなりません（ただし近い比率にはなる）。また、エンベロープによっては、あまり小さなG／S値に設定すると、発音しても聞き取れないことがあります。

本コマンドは、次のようなディレイド機能フォーマットでも実現できます。

第1バイト：　82H
第2バイト：　G／S値

○ NOTE ON

機　能　音程／音長の設定
入　力　AH←13H
　　　　AL←指定チャンネル
　　　　BH←キー番号
　　　　BL←音長（ステップタイム）
出　力　なし

本コマンドによって、指定される音程、音長に従って発音します。音程に与えることのできる値の範囲は0から96の値で、オクターブと音名の対応は表5－4のようになっています。

	O1	O2	O3	O4	O5	O6	O7	O8	O9
C	0	C	18	24	30	3C	48	54	60
C♯ (D♭)	1	D	19	25	31	3D	49	55	
D	2	E	1A	26	32	3E	4A	56	
D♯ (E♭)	3	F	1B	27	33	3F	4B	57	
E	4	10	1C	28	34	40	4C	58	
F	5	11	1D	29	35	41	4D	59	
F♯ (G♭)	6	12	1E	2A	36	42	4E	5A	
G	7	13	1F	2B	37	43	4F	5B	
G♯ (A♭)	8	14	20	2C	38	44	50	5C	
A	9	15	21	2D	39	45	51	5D	
A♯ (B♭)	A	16	22	2E	3A	46	52	5E	
B	B	17	23	2F	3B	47	53	5F	

（表5-4　キー番号と音名の対応表）

本コマンドを用いて発音した場合、前の音がまだ発音中である場合には、前の音の発音を切って新しい音を発音します。よって、連続的に発音を行う場合には、ディレイド機能フォーマットを用いた方が望ましいといえます。

本コマンドのディレイド機能フォーマットは、次のようになっています。

第1バイト：　キー番号（0から60H，80Hは休符）

第2バイト：　音長（00H：規定値、01HからFFH：ステップタイム）

音長はステップタイムで指定します。通常は、1小節あたりのステップ数で演奏のきめこまかさを設定しますが、1小節96ステップタイムで十分で、192ステップタイムもあれば完璧といえます。ステップタイムに対する演奏速度は、テンポ設定で行います。1小節を192ステップタイムとすれば、4分音符は48ステップタイムとなり、64分音符の3連符までが表現できるわけです。

○ SET DEFAULT LENGTH

機　能　音長のデフォルト値を設定する

入　力　AH ← 14 H

AL ←指定チャンネル

BL ←音長

出　力　なし

本コマンドは、音長が省略された場合のデフォルト値を、ステップタイムで設定するものです。NOTE コマンドなどで音長が省略された場合には、本コマンドによる値が採用されます。本コマンドに対応するディレイド機能フォーマットは次のとおりです。

第1バイト：　83H

第2バイト：　音長（01H～FFH）

○ SET TEMPO

機　能　テンポを設定します

入　力　AH ← 15 H

BL ←テンポ数

出　力　なし

本コマンドは、全チャンネル共通のテンポ数を設定するものです。テンポ数の単位は、1秒

間に48ステップを何回演奏できるかということです。たとえば、ある曲が全音符（4／4）を4秒かかって演奏するとし、テンポ数を192ステップタイム（04H）とした場合には、48ステップタイムは4分音符1個の計算になり、4分音符＝4とテンポ数に一致するわけです。本コマンドに対応するディレイド機能フォーマットは次のとおりです。

第1バイト：　84H
第2バイト：　テンポ数（01H～FFH）

○ SET PARAMETER BLOCK

機　能　各チャンネルに対する音色パラメータの設定
入　力　AH ← 16 H
　　　　AL ←指定チャンネル
　　　　ES:BX ←音色パラメータブロックのアドレス
　　　　DL ←パラメータブロックの形式
出　力　なし

本コマンドは、各チャンネルに対する音色パラメータを設定します。レジスタ DL に対する値が0のときには、パラメータブロックはワード配列となり、各パラメータはワード単位でアクセスされます（サイズは100バイト）。また、レジスタ DL に対する値が1であるときには、パラメータブロックはバイト配列（ワードデータはワードのまま）となり、各パラメータはバイト単位でアクセスされます（サイズは51バイト）。指定されたチャンネルが FM 音源でない場合には、設定は無視されます。

本コマンドに対するディレイド機能フォーマットは、次のようになっています。

第1バイト　　　　：　85H
第2バイト　　　　：　パラメータブロック形式
第3，第4バイト：　パラメータブロックオフセット
第5，第6バイト：　パラメータブロックセグメント

また、パラメータブロックの形式は、5．1で説明した音色データのフォーマットに準拠します。ただし、データブロックの最高尾に、以下の情報が付け加わります。

オフセット33H：　キーステータスセーブエリア（作業用）

○ READ PARAMETER

機　能　　各チャンネルの音色パラメータを読み出す

入　力　　AH ← 17 H

AL ←指定チャンネル

BL ←パラメータ番号

出　力　　BX ←設定値

本コマンドは、各チャンネルに設定されている音色データのうち、1個の任意のパラメータを読み出します。パラメータ番号は、0からの値で、下位アドレスに位置するものから順に付けられています。

○ WRITE OPERATER

機　能　　各チャンネルに音色パラメータを設定する

入　力　　AH ← 18 H

AL ←指定チャンネル

BL ←パラメータ番号

DX ←設定値

出　力　　なし

本コマンドは、各チャンネルに対して、音色データを1個の任意のパラメータについて設定します。本コマンドにおけるディレイド機能フォーマットは次のようになっています。

第1バイト　　　：　86H

第2バイト　　　：　パラメータ番号

第3，第4バイト：　設定値

○ ALL STOP

機　能　　演奏を一時的に中断する

入　力　　AH ← 19 H

出　力　　なし

本コマンドは、現在実行中の演奏を中断します。中断した演奏は、CONTINUE PLAY コマンドによって再開することができます。

○ CONTINUE PLAY

機　能　ALL STOPによって中断した演奏を再開する
入　力　AH←1AH
出　力　なし

ALL STOPコマンドによって中断されていた演奏を再開させます。ALL STOPコマンドが実行されていないのに本コマンドを実行しても支障はありません。

○ HOLD STATUS

機　能　キーの状態を保持します
入　力　AH←1EH
　　　　AL←指定チャンネル
　　　　BL←維持する長さ
出　力　なし

本コマンドは、現在のキーの音／オフの状態を保持したまま、指定するステップタイム数待ちます。WRITE REGISTERコマンドなどでキーの音／オフを設定したのちに、その状態を保持したい場合に、本コマンドを用います。

本コマンドに対するディレイド機能フォーマットは、次のようになっています。

第1バイト：　89H
第2バイト：　維持する長さ（01H～FFH）

○ MODU ON

機　能　LFO効果のON
入　力　AH←1BH
　　　　AL←指定チャンネル
出　力　なし

本コマンドは、音色に付加されるLFO効果を開始させるためのものです。

本コマンドに対するディレイド機能フォーマットは、次のようになっています。

第1バイト：　87H

○ MODU OFF

機　能　　ＬＦＯ効果のＯＦＦ

入　力　　AH ← 1CH

　　　　　AL ←指定チャンネル

出　力　　なし

本コマンドは、音色に付加されるＬＦＯ効果を停止させます。

本コマンドに対するディレイド機能フォーマットは、次のようになっています。

第1バイト：　８８Ｈ

○ SET INTERRUPT CONDITION

機　能　　ＰＬＡＹバッファエンプティ割り込みの条件を設定する

入　力　　AH ← 1DH

　　　　　AL ←指定チャンネル

　　　　　ES:BX ←割り込み処理ルーチン開始アドレス

　　　　　CX ←条件

出　力　　なし

本コマンドは、ＰＬＡＹバッファ内の有効データ数によって、割り込みを発生させる条件を設定するためのものです。条件はレジスタ CX に、制御情報域に格納されるものと同じものを与えます。また、割り込み処理ルーチンのアドレスはレジスタ ES:BX に与え、割り込み処理ルーチンはセグメント外 RET で終結させます。これは、割り込み処理ルーチンが、セグメント外 CALL によって呼び出されるからです。

割り込み処理中は、必ず外部割り込みを禁止（IF = 1）としておかなくてはなりません。また、割り込み処理にかかる時間は、できるだけ短いものでなくてはなりません。あまり長いと、演奏に支障がでる場合があります。

割り込み処理ルーチンが呼び出された際のレジスタの内容は次のようになっています。

レジスタ AX：　演奏中チャンネル番号

レジスタ BX：　演奏中バッファ有効バイト数

○ SET VOLUME

機　能　　ＦＭ音源の音量を設定する

入　力　　AH ← 1AH

AL ←指定チャンネル
BL ←音量設定値
出　力　なし

本コマンドは、FM音源の音量を設定するためのものです。音量設定値は、００Hが最小、７FHが最大となっており、制御情報域のアルゴリズムを参照し、出力オペレータ（キャリア）の出力レベルを操作します。

本コマンドに対するディレイド機能フォーマットは、次のようになっています。

第1バイト：　8AH
第2バイト：　音量設定値

付録C　無限音階

いわゆる音階とは意味が異なりますが、無限音階というものがあります。通常の音階は、CDEFG…と音程を上げていくと次第に高い音になり、ついには可聴周波数の上限を越えて聞こえなくなってしまいます。これに対して無限音階は、次第に高くなる感じはするものの、聞こえなくなってしまうことはなく、いつまでも繰り返されるものです。

図5－5を見てください。通常の楽音の倍音構成をみると、(a)のように、音程を決める基本音が存在し、オクターブ4のC音とオクターブ5のC音とを区別することができます。一方無限音階では、(b)のように可聴周波数全域にわたって倍音が均等に分布しているため、どれが基本音であるか区別がつかなくなります。このような倍音構成をもつ音をC～Bまで12種類用意すれば無限音階を構成することができます。

オクターブの概念がなくなってしまうのですから、たとえばCとGの音はどちらが高いとも低いともいえません。しかし、聴感上、CDEFG…と順に発音すれば音が高くなっていくように聞こえるのです。

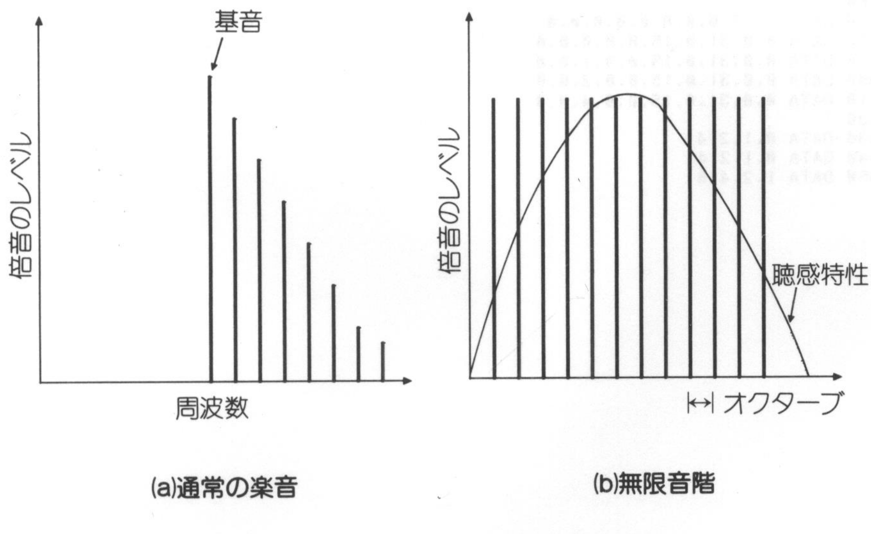


図5-5　倍音構成

ＯＰＮには、12個のオペレータがあり、すべてをキャリアとしてもちいると12オクターブにわたる倍音構成を得ることができ、これでほぼ可聴周波数をカバーできるので無限音階を構成することができます。サンプルとして次のプログラムを掲げておきます。

```
1000 ' 無限音階
1010 'prog92.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256,256,256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4,9)
1080 VOICE COPY 0,V
1090 '
1100 FOR I=0 TO 4
1110   FOR J=0 TO 9
1120     READ V(I,J)
1130   NEXT
1140 NEXT
1150 '
1160 FOR I=1 TO 3
1170   FOR J=1 TO 4
1180     READ V(J,7)
1190   NEXT
1200   VOICE I,V
1210 NEXT
1220 '
1230 PLAY "@1V15","@2V15","@3V15"
1240 PLAY "O1ABCDEFG","O5ABCDEFG","O8ABCDEFG"
1250 GOTO 1240
1260 '
1270 DATA 7,15,0,0,0,0,0,0,0,0
1280 DATA 0,0,31,0,15,0,0,0,0,0
1290 DATA 0,0,31,0,15,0,0,1,0,0
1300 DATA 0,0,31,0,15,0,0,2,0,0
1310 DATA 0,0,31,0,15,0,0,4,0,0
1320 '
1330 DATA 0,1,2,4
1340 DATA 0,1,2,4
1350 DATA 1,2,4,8
```

付録D　SensとDepthの相違

ＬＦＯのパラメータには、ピッチモジュレーションについてＰＭＳとＰＭＤが、アンプリチュードモジュレーションについてＡＭＳとＡＭＤがあります。Sens と Depth はどちらも変調量を決めるパラメータですが、どのように違うのでしょうか。

ＲＯＭ内ルーチンを解析したところ、実際の変調量はDepthの値にSensを２の指数として乗じたものが用いられています。つまり変調量は、

$$Depth * 2^{Sens}$$

で表わされます。ただし、Sens＝０のときは変調量は０となります。

付録E　オペレータによるビブラート

ＯＰＮにはＬＦＯが内蔵されていませんので、ＰＣシリーズではＬＦＯ効果を得るためにタイマの割込みを利用し、ソフトウェアで行なっています。ただし、トレモロ効果(音量変調)、ワウ効果（音色変調）については、ＳＳＧ―ＥＧ機能を用いて代用することができるのですが、ここではビブラート効果（周波数変調）をハードウェア的に得る方法について解説します。

ビブラートとは、音の周波数が微妙に変化することをいいます。次の図を見てください。

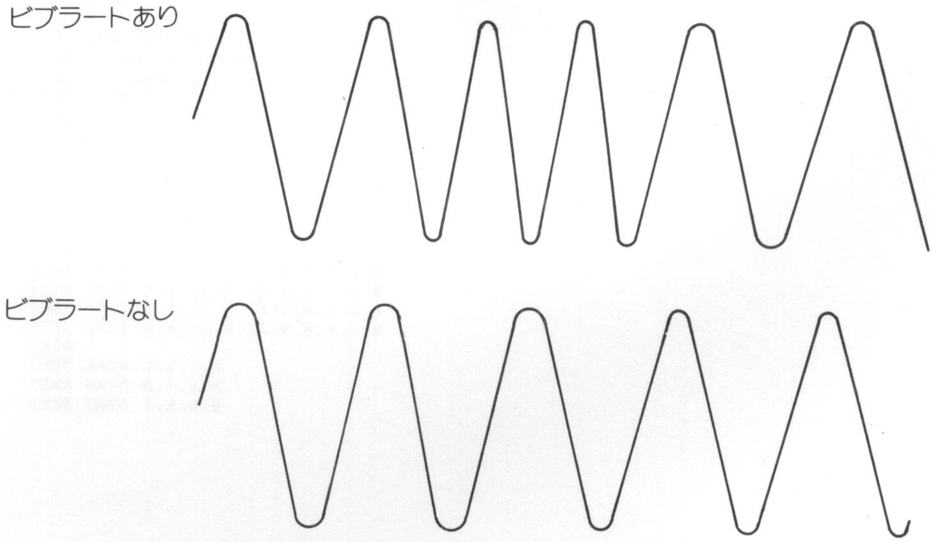


図5-6　ビブラート

ここで、図１－１(p.3)をもういちど見てください。上図のビブラートのかかった状態は、図１－１（ｂ）とまったく同じです。つまり、モジュレータの周波数をキャリアの周波数に対して（波形に変化を与えないほど）充分低く設定すれば、自然にビブラートがかかるわけです。

次に示すサンプルは、音色番号7 FLUTE にビブラートをかけています。この音色にはソフトウェアＬＦＯがかけてありますので、まず、PMDを0にしてこれをマスクしています。続いて、モジュレータとなるOp 3のMlを0、キャリアとなるOp 4のMlを15に設定しています。この結果、モジュレータとキャリアの周波数比は、１：30となります。

マルチプルを使うこの方法では、ビブラート周波数が音の高さに依存しますが、チャンネル3で効果音モードを使用すれば独立に設定することも可能です。

```
1000 ' フルートサウンドにOPNでビブラートをかける
1010 'prog93.bas",a
1020 CLEAR &H300
1030 PLAY ALLOC 256.256.256
1040 OPTION BASE 0
1050 DEFINT A-Z
1060 VOICE INIT
1070 DIM V(4.9)
1080 VOICE COPY 7,V
1090 '
1100 V(0.5)=0
1110 V(3.7)=0
1120 V(4,7)=15
1130 VOICE 7.V
1140 '
1150 PLAY "O7L2O1E1"
1160 END
```

オリジナルボイス　データシート

（コピーしてお使いください）

oct ～

	Op1	Op2	Op3	Op4	
Fb/AI	(/)				→ v(0,0)
Dt/MI	(/)	(/)	(/)	(/)	Dt → v(OP,8) MI → v(OP,7)
OI					→ v(OP,5)
KS/AR	(/)	(/)	(/)	(/)	KS → v(OP,6) AR → v(OP,0)
DR					→ v(OP,1)
SR					→ v(OP,2)
SL/RR	(/)	(/)	(/)	(/)	SL → v(OP,4) RR → v(OP,3)
AMS					→ v(OP,9)
Wave					→ v(0,2)
Sync					→ v(0,3)
Speed					→ v(0,4)
PMD					→ v(0,5)
AMD					→ v(0,6)
PMS					→ v(0,7)

エンベロープ

Op1	Op2	Op3	Op4

○音色イメージなど

PC-9800シリーズ
FM音源スーパーサウンド

♪ 共著 窪田 弘
田中寿郎
広瀬 真

♪ 発行者 牧谷秀昭
♪ 発行所 秀和システムトレーディング株式会社
〒107 東京都港区赤坂8—5—29 チエッカービル
♪ 印刷所 共同印刷株式会社

♪ 発行日 1986年12月26日 初版第1刷
1987年11月28日 初版第2刷

ISBN4-87966-105-8 C3055 ¥1900E

秀和システムトレーディング株式会社　定価 1,900円

ISBN4-87966-105-8 C3055 ¥1900E